

TRABAJO DE FIN DE GRADO

**Grado en Ingeniería eléctrica**

**ESTUDIO Y ANÁLISIS DEL CONSUMO ELÉCTRICO DE  
EDIFICIOS DE USO UNIVERSITARIO**



**Memoria y Anexos**

<b>Autor:</b>	Francisco Javier Pereira Martínez
<b>Director:</b>	Sergi Fillet Castella
<b>Codirectora:</b>	Carmen Ortiz Caraballo
<b>Convocatoria:</b>	Abril 2019



## Resum

El present treball es centra en l'estudi de senyals de consum elèctric d'un edifici universitari mitjançant unes funcions anomenades *wavelets*. Es presentaran diverses d'elles, encara que s'utilitzarà en tot moment la wavelet de Haar, que és la més simple i antiga de totes les que existeixen.

Primerament s'exposaran les bases teòriques i conceptuals de com treballen les wavelets i què poden arribar a fer. El segon bloc del treball consistirà en estudiar en una dualitat freqüència-temps senyals diàries de consum elèctric mitjançant les transformades discreta i contínua de wavelet, on es podrà observar el contingut freqüencial existent en cada moment al llarg de tota la senyal. S'utilitzarà tant l'app *waveletAnalyzer* que proporciona Matlab com comandaments en la *command window* per a realitzar l'estudi.

Finalment s'exposarà un model de predicció, acompanyat per un algoritme desenvolupat per a Matlab, que permetrà predir el consum d'un dia concret tenint les dades del dia anterior i un històric.

Les dades que s'utilitzen són les de l'escola ETSEIB de la Universitat Politècnica de Catalunya de l'any 2014 en gran part. Per a l'algoritme és utilitzat un dia de l'any 2015 amb tot l'any 2014 com a històric.

## Resumen

El presente trabajo se centra en el estudio de señales de consumo eléctrico de un edificio universitario mediante unas funciones llamadas wavelets. Se presentarán varias de ellas, aunque se utilizará en todo momento la wavelet de Haar, que es la más simple y antigua de todas las que existen.

Primeramente, se expondrán las bases teóricas y conceptuales de cómo trabajan las wavelets y que pueden llegar a hacer. El segundo bloque del trabajo consistirá en estudiar en una dualidad frecuencia-tiempo señales diarias de consumo eléctrico mediante las transformadas discreta y continua de wavelet, donde se podrá observar el contenido frecuencial existente en cada momento a lo largo de toda la señal. Se utilizará tanto la app *waveletAnalyzer* que proporciona Matlab como mandos en la *command window* para realizar el estudio.

Finalmente se expondrá un modelo de predicción, acompañado por un algoritmo desarrollado para Matlab, que permitirá predecir el consumo de un día concreto teniendo los datos del día anterior y un histórico.

Los datos que se utilizan son las de la escuela ETSEIB de la Universidad Politécnica de Cataluña del 2014 en gran parte. Para el algoritmo es utilizado un día del año 2015 con todo el año 2014 como histórico.

## Abstract

The present work focuses on the study of electrical consumption signals of a university building through functions called wavelets. Several of them will be presented, although we will use the Haar wavelet all the time, which is the simplest and oldest of all those that exist.

First, the theoretical and conceptual bases of how wavelets work and what they can do will be explained. The second block of the work will consist in studying in a frequency-time duality daily signals of electrical consumption by means of the discrete and continuous wavelet transforms, where the frequency content existing in each moment can be observed along the signal. Both the *waveletAnalyzer* app provided by Matlab and commands in the *command window* will be used to carry out the study.

Finally, a prediction model will be exposed, accompanied by an algorithm developed for being used in Matlab, which will allow predicting the consumption of a specific day taking the data of the previous day and the data of the whole previous year.

The data used are those of the ETSEIB school of the Polytechnic University of Catalonia in 2014 to a large extent. For the algorithm it is used one day of the year 2015 with the whole year 2014 as historical.



# Índice

<b>RESUM</b>	<b>I</b>
<b>RESUMEN</b>	<b>II</b>
<b>ABSTRACT</b>	<b>III</b>
<b>1. PREFACIO</b>	<b>1</b>
1.1. Origen del trabajo	1
1.2. Motivación	1
<b>2. INTRODUCCIÓN</b>	<b>3</b>
2.1. Objetivos del trabajo	3
2.2. Alcance del trabajo	3
<b>3. INTRODUCCIÓN A LAS WAVELETS</b>	<b>5</b>
3.1. Qué es una wavelet y cómo trabaja	6
3.2. Haar	11
3.3. Breve historia	13
3.4. Utilidad en el análisis de datos	13
3.5. Presentación y explicación de las herramientas de Matlab que se van a utilizar	14
3.5.1. Estudio de la transformada mediante los display mode que ofrece Wavelet 1-D	14
3.5.2. Wavelet Coefficients Selection 1-D	18
3.5.3. Continuous Wavelet 1-D	19
3.5.4. Wavelets packets 1-D	19
<b>4. EXPOSICIÓN DE LOS DATOS Y PRESENTACIÓN DE LAS CARACTERÍSTICAS DE LOS EDIFICIOS A ESTUDIAR.</b>	<b>21</b>
4.1. Calendarios laborales de Barcelona (2014-2017)	21
4.1.1. Festivos propios de la universidad	25
4.2. Datos recogidos	26
4.2.1. Año 2014	26
4.2.2. Año 2015	26
4.2.3. Año 2016	27
4.2.4. Año 2017	28
<b>5. ANÁLISIS FRECUENCIA-TIEMPO DE LOS DATOS</b>	<b>30</b>

5.1.	Explicación previa .....	30
5.2.	Procedimiento de análisis.....	31
5.2.1.	Descomposición de las señales originales.....	31
5.2.2.	De-noising de las señales originales .....	34
5.2.3.	Descomposición de las señales sin ruido.....	37
5.2.4.	Transformada continua de Wavelet (CWT).....	39
5.3.	Datos obtenidos y conclusiones .....	40
5.3.1.	CWT de las señales sin descomponer.....	40
5.3.2.	CWT de los coeficientes de detalle de nivel 1 (cd1r).....	43
5.3.3.	CWT de los coeficientes de detalle de nivel 2 (cd2r).....	46
5.3.4.	CWT de los coeficientes de aproximación de nivel 2 (a2r) .....	49
5.3.5.	CWT de la suma de los coeficientes de detalle (cd1r+cd2r) .....	50
5.3.6.	Valoraciones finales .....	52
<b>6.</b>	<b>MODELIZACIÓN PARA LA PREDICCIÓN DE CONSUMOS .....</b>	<b>54</b>
6.1.	Explicación del modelo predictivo a utilizar.....	54
6.2.	Procedimiento de predicción .....	55
6.3.	Datos obtenidos y conclusiones .....	56
<b>7.</b>	<b>PROPUESTAS DE CONTINUIDAD DEL TRABAJO .....</b>	<b>57</b>
	<b>CONCLUSIONES .....</b>	<b>58</b>
	<b>PRESUPUESTO .....</b>	<b>59</b>
	<b>BIBLIOGRAFÍA .....</b>	<b>60</b>
	<b>ANEXO A .....</b>	<b>63</b>
A1.	Comandos Matlab .....	63
A2.	Algoritmo de predicción .....	67



## Índice de figuras

Figura 3.1. Wavelet db2 (Fuente: Wavelet display de Matlab)	6
Figura 3.2. Diferentes tipos de wavelets (de izquierda a derecha y de arriba a abajo: db8, sym3, coif5, fk4) (Fuente: Wavelet display de Matlab)	8
Figura 3.3. Proceso de descomposición de la DWT (Fuente: Wikipedia: Discrete Wavelet Transform)	10
Figura 3.4. Descomposición de una señal S en 4 niveles mediante la wavelet de Haar (Fuente: Elaboración propia)	11
Figura 3.5. Wavelet de Haar (Fuente: Elaboración propia)	11
Figura 3.6. Dilataciones y translaciones de la wavelet de Haar en [0,1] (Fuente: Wavelets for kids)	12
Figura 3.7. Señal 'noischir'. (Fuente: Matlab)	14
Figura 3.8. Coeficientes de los niveles 1-6 de la señal 'noischir' (Fuente: waveletAnalyzer de Matlab)	15
Figura 3.9. Descomposición en 5 niveles de la señal 'noischir' en 'Separate Mode' (Fuente: waveletAnalyzer de Matlab)	15
Figura 3.10. Ejemplo de una señal sin sus coeficientes más irrelevantes (Fuente: waveletAnalyzer de Matlab)	16
Figura 3.11. Selección de 256 coeficientes de los 1024 totales (Fuente: waveletAnalyzer de Matlab)	19
Figura 3.12. Descomposición de una señal usando un paquete wavelet (Fuente: waveletAnalyzer de Matlab)	20
Figura 3.13. Wavelet Packets 1-D con la señal del 15 de enero de 2014 (Fuente: waveletAnalyzer de Matlab)	20
Figura 4.1. Primer semestre del calendario laboral de Barcelona en 2014 (Fuente: <a href="https://www.calendarioslaborales.com/">https://www.calendarioslaborales.com/</a> )	21

Figura 4.2. Segundo semestre del calendario laboral de Barcelona en 2014 (Fuente: <a href="https://www.calendarioslaborales.com/">https://www.calendarioslaborales.com/</a> )	22
Figura 4.3. Primer semestre del calendario laboral de Barcelona en 2015 (Fuente: <a href="https://www.calendarioslaborales.com/">https://www.calendarioslaborales.com/</a> )	22
Figura 4.4. Segundo semestre del calendario laboral de Barcelona en 2015 (Fuente: <a href="https://www.calendarioslaborales.com/">https://www.calendarioslaborales.com/</a> )	23
Figura 4.5. Primer semestre del calendario laboral de Barcelona en 2016 (Fuente: <a href="https://www.calendarioslaborales.com/">https://www.calendarioslaborales.com/</a> )	23
Figura 4.6. Segundo semestre del calendario laboral de Barcelona en 2016 (Fuente: <a href="https://www.calendarioslaborales.com/">https://www.calendarioslaborales.com/</a> )	24
Figura 4.7. Primer semestre del calendario laboral de Barcelona en 2017 (Fuente: <a href="https://www.calendarioslaborales.com/">https://www.calendarioslaborales.com/</a> )	24
Figura 4.8. Segundo semestre del calendario laboral de Barcelona en 2017 (Fuente: <a href="https://www.calendarioslaborales.com/">https://www.calendarioslaborales.com/</a> )	25
Figura 4.9. Festivos generales ETSEIB 2014-2017 (Fuente: <a href="https://www.upc.edu/sga/es/normativas/CalendariosAcademicos">https://www.upc.edu/sga/es/normativas/CalendariosAcademicos</a> )	25
Figura 4.10. Consumo eléctrico ETSEIB 2014 en kWh	26
Figura 4.11. Consumo eléctrico ETSEIB 2015 en kWh	27
Figura 4.12. Consumo eléctrico ETSEIB 2015 en kWh (aumentado)	27
Figura 4.13. Consumo eléctrico ETSEIB 2016 en kWh	27
Figura 4.14. Consumo eléctrico ETSEIB 2016 en kWh (aumentado)	28
Figura 4.15. Consumo eléctrico ETSEIB 2017 en kWh	28
Figura 4.16. Consumo eléctrico ETSEIB 2017 en kWh (aumentado)	29
Figura 5.1. Consumo (kWh) diario ETSEIB 2014	31
Figura 5.2. Descomposición con Haar de la señal del 15 de enero de 2014 en dos niveles	32
Figura 5.3. Descomposición con Haar de la señal del 9 de abril de 2014 en dos niveles	32

Figura 5.4. Descomposición con Haar de la señal del 13 de agosto de 2014 en dos niveles	33
Figura 5.5. Descomposición con Haar de la señal del 31 de diciembre de 2014 en dos niveles	33
Figura 5.6. Thresholding aplicado a la señal del 15 de enero de 2014 (HS)	34
Figura 5.7. Thresholding aplicado a la señal del 9 de abril de 2014 (HS)	34
Figura 5.8. Thresholding aplicado a la señal del 13 de agosto de 2014 (HS)	34
Figura 5.9. Thresholding aplicado a la señal del 13 de agosto de 2014 (manual)	35
Figura 5.10. Thresholding aplicado a la señal del 31 de diciembre de 2014 (HS)	35
Figura 5.11. Thresholding aplicado a la señal del 31 de diciembre de 2014 (manual)	35
Figura 5.12. Residuales del thresholding aplicado a la señal del 15 de enero de 2014 (HS)	36
Figura 5.13. Residuales del thresholding aplicado a la señal del 9 de abril de 2014 (HS)	36
Figura 5.14. Residuales del thresholding aplicado a la señal del 13 de agosto de 2014 (HS)	36
Figura 5.15. Residuales del thresholding aplicado a la señal del 13 de agosto de 2014 (manual)	36
Figura 5.16. Residuales del thresholding aplicado a la señal del 31 de diciembre de 2014 (HS)	37
Figura 5.17. Residuales del thresholding aplicado a la señal del 31 de diciembre de 2014 (manual)	37
Figura 5.18. Descomposición S2014enero15d	38
Figura 5.19. Descomposición S2014abril9d	38
Figura 5.20. Descomposición S2014agosto13d	39
Figura 5.21. Descomposición S2014diciembre31d	39
Figura 5.22. CWT(S2014enero15d) en 3D y en 2D	40
Figura 5.23. CWT(S2014abril9d) en 3D y en 2D	41
Figura 5.24. CWT(S2014agosto13d) en 3D y en 2D	41
Figura 5.25. CWT(S2014diciembre31d) en 3D y en 2D	41

Figura 5.26. CWT(S2014enero15d-cd1r) en 3D y en 2D	43
Figura 5.27. CWT(S2014abril9d-cd1r) en 3D y en 2D	43
Figura 5.28. CWT(S2014agosto13d-cd1r) en 3D y en 2D	44
Figura 5.29. CWT(S2014diciembre31d-cd1r) en 3D y en 2D	44
Figura 5.30. CWT(S2014enero15d-cd2r) en 3D y en 2D	46
Figura 5.31. CWT(S2014abril9d-cd2r) en 3D y en 2D	46
Figura 5.32. CWT(S2014agosto13d-cd2r) en 3D y en 2D	47
Figura 5.33. CWT(S2014diciembre31d-cd2r) en 3D y en 2D	47
Figura 5.34. CWT(S2014enero15d-a2r) en 3D y en 2D	49
Figura 5.35. CWT(S2014abril9d-a2r) en 3D y en 2D	49
Figura 5.36. CWT(S2014agosto13d-a2r) en 3D y en 2D	49
Figura 5.37. CWT(S2014diciembre31d-a2r) en 3D y en 2D	50
Figura 5.38. CWT(S2014enero15d-cd1r+cd2r) en 3D y en 2D	50
Figura 5.39. CWT(S2014abril9d-cd1r+cd2r) en 3D y en 2D	51
Figura 5.40. CWT(S2014agosto13d-cd1r+cd2r) en 3D y en 2D	51
Figura 5.41. CWT(S2014diciembre31d-cd1r+cd2r) en 3D y en 2D	51
Figura 6.1. Consumo obtenido y consumo real (Fuente: Elaboración propia)	56
Figura 6.2. Semanas de los días más parecidos según tipo de coeficientes	56
Figura 6.3. Valor eficaz de la señal real y de la predicción	56





# 1. Prefacio

## 1.1. Origen del trabajo

Cuando mi tutor me presentó este proyecto y decidimos llevarlo a cabo también me explicó que había un precedente. El curso anterior se había realizado, también como trabajo final de grado, un estudio de estos mismos datos utilizando propiedades de las series temporales. En ese trabajo, llamado *Disseny d'un model per l'estimació del consum elèctric i la seva implementació en R<sup>1</sup>*, por Marta Garzón Àngel, en el que se realizaba el estudio del consumo desde el punto de vista clásico de las series temporales, se llegaba a la conclusión de que el comportamiento de la serie de datos de consumo que se estudian no puede describirse con exactitud con los métodos de análisis de las series temporales, ya que no quedaban del todo bien reflejados ciertos parámetros de la serie, tales como la variación de la tendencia entre días de la semana y fines de semana, o la falta de presencia clara de periodos no lectivos. Tampoco se pudo extraer un patrón preciso de consumo diario o semanal, con lo cual se determinaba que las series temporales no eran el mejor método de analizar este tipo de datos.

Mi tutor me ofreció la posibilidad de realizar un estudio de los datos de consumo de edificios de uso universitario utilizando unas funciones llamadas wavelets, que serán explicadas más adelante, y que tienen la particularidad de trabajar en un entorno frecuencia-tiempo. Habiendo visto que un estudio en el entorno temporal no es la mejor herramienta para el análisis de estos datos, este trabajo se convertía en una primera aproximación para ver si un estudio frecuencia-tiempo utilizando wavelets es más preciso.

## 1.2. Motivación

Tenía claro que quería realizar un trabajo relacionado con Matlab, ya que considero que es una herramienta muy potente de la que tenía conocimientos básicos, y de este modo adquirir más soltura y conocimiento sobre este programa que sin duda me será útil en el futuro. Además, el hecho de elaborar una continuación a otro estudio realizado previamente me pareció muy interesante. Por otro lado, el uso de las wavelets se planteaba como un reto para demostrar mi capacidad de autoaprendizaje de conceptos no vistos previamente durante la carrera, así que supe que tenía que dedicar un esfuerzo adicional en conocer cómo funcionaban exactamente y profundizar en las herramientas de análisis frecuencial tan utilizadas en el sector eléctrico.





## 2. Introducción

En este trabajo se explica la teoría de wavelet y su aplicación en el análisis de datos, así como su uso como herramienta de predicción. Se aplica en datos de consumo eléctrico de la universidad ETSEIB perteneciente a la UPC.

Guarda relación con el trabajo realizado por Marta Garzón Àngel<sup>1</sup> en julio de 2018 desde el momento que se trabaja con los mismos datos iniciales y tiene muchos objetivos comunes, aunque la metodología sea totalmente distinta.

### 2.1. Objetivos del trabajo

El objetivo principal es el de desarrollar la metodología de aplicación de la transformada de wavelet para localizar patrones de comportamiento de unos datos de consumo eléctrico de una universidad en el tiempo y separar sus componentes frecuenciales.

En particular, se pretende determinar si mediante la wavelet de Haar es posible hacer un estudio de los datos de consumo eléctrico recogidos cada 15 minutos en una universidad o si por el contrario se obtienen resultados no óptimos y habrá que probar con otro tipo de wavelet o señales de diferente frecuencia de muestreo.

El objetivo final será el de crear un algoritmo de predicción que nos permita conocer el consumo de un día completo de esta universidad utilizando la descomposición de la transformada de wavelet.

### 2.2. Alcance del trabajo

El alcance de este trabajo empieza con la comprensión de la transformada de wavelet, los conceptos que tiene asociados, tales como *niveles* y *escalas*, y las herramientas relacionadas con ella. Se utilizarán las herramientas que se consideren oportunas y se usará la wavelet de Haar en todo momento para analizar las señales de consumo y extraer conclusiones.

El trabajo termina con el desarrollo de un algoritmo predictor que sirva como base para la predicción de consumos basándose en el proceso de descomposición de señales que hace la transformada de wavelet e inspirado en el algoritmo de los *k vecinos más cercanos* (k-nearest neighbors).

Si se concluye que la metodología es depurable, se propondrán propuestas de mejora en el proceso seguido, tanto para el estudio de las señales como para la predicción, y vías de continuación del trabajo.



### 3. Introducción a las Wavelets

A menudo nos encontramos con funciones complejas que no son fáciles de analizar, compuestas por miles de datos que no permiten un tratamiento de conjunto, sino que hay que ir analizándola por partes y, dada su magnitud, acaban por no tener ninguna utilidad práctica.

Es por ello que se utilizan técnicas de regresión y aproximación a dichas funciones, de modo que nos sea más fácil ver lo que muestra una serie de datos en su totalidad. Como ejemplos, tenemos la serie de Taylor [1], que aproxima una función utilizando una serie de potencias, o la serie de Fourier [2], basada en la descomposición de dicha función en una suma infinita de funciones sinusoidales mucho más simples.

$$\sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!} (x - a)^n \quad [1]$$

Donde

- $n!$  es el factorial de  $n$ .
- $f^{(n)}(a)$  es la  $n$ -ésima derivada de  $f$  en el punto  $a$ .

$$\frac{a_0}{2} + \sum_{n=1}^{\infty} \left[ a_n \cos\left(\frac{2n\pi}{T} t\right) + b_n \sin\left(\frac{2n\pi}{T} t\right) \right] \quad [2]$$

Donde  $a_n$  y  $b_n$  se denominan coeficientes de Fourier.

La descomposición de Fourier es utilizada ampliamente para el estudio de señales eléctricas, permitiendo identificar comportamiento frecuencial pero perdiendo en el camino la información sobre en qué instantes dentro del periodo analizado se hayan estas componentes. Por lo tanto, presenta problemas para utilizarse en el análisis de señales con periodicidades que no están presentes continuamente (apareciendo, por ejemplo, solo durante una pequeña parte de la señal analizada).

Cuando trabajamos con una señal, frecuentemente, nos interesa conocer aspectos de su comportamiento frecuencial y temporal. La transformada de Fourier no nos da toda la información que queremos, ya que se obtiene gran resolución frecuencial, pero se pierde toda información temporal. Las series temporales tampoco (las herramientas de análisis basadas en la respuesta temporal presentan el problema inverso).

En este trabajo se utilizarán otra clase de funciones, que no son ni polinomios ni sinusoidales, y cuyo nombre en inglés es wavelets. Etimológicamente, la palabra Wavelet sugiere que es una función finita y está bien localizada y acotada<sup>2</sup>, como podemos ver en la figura 3.1. Coloquialmente, se podría traducir wavelet como “onda pequeña”. En el trabajo utilizaremos la tradicional denominación en inglés por estar ampliamente establecida.

La transformada de wavelet se acerca mucho al objetivo buscado, que es trabajar en ambos dominios (frecuencia-tiempo).

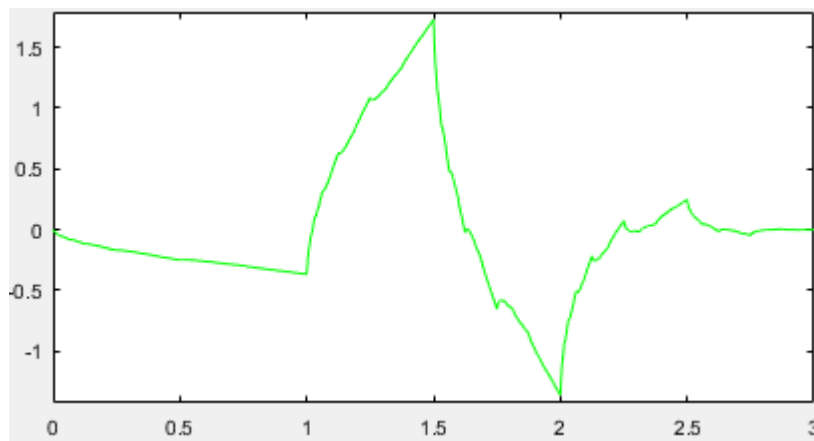


Figura 3.1. Wavelet db2 (Fuente: Wavelet display de Matlab)

Con propósito de situarnos en el contexto adecuado, y para tener una fotografía general de lo que se va a trabajar en este documento, se intenta a continuación resumir brevemente el concepto de Wavelet y su posicionamiento en el ámbito científico como herramienta a tener muy en cuenta.

### 3.1. Qué es una wavelet y cómo trabaja

Nos situamos en que tenemos una función no estacionaria. Si nos fijamos en una parte en concreto de la señal, podemos analizar el contenido frecuencial en un periodo de tiempo concreto. La resolución temporal dependerá de qué pequeño sea el trozo que observamos, ya que si es relativamente pequeño sabremos qué está pasando en una zona de tiempo muy concreta, aunque también estaremos perdiendo información de frecuencia al analizar ese trozo tan acotado de la señal. En el caso contrario pasa al revés, en un trozo de señal amplio tendremos toda clase de frecuencias, pero no sabremos en qué momento están ocurriendo<sup>3</sup>. Podemos decir que la resolución temporal y la resolución del contenido de frecuencia son inversamente proporcionales.

Con las wavelets podemos definir cómo queremos analizar la señal: mediante dilataciones controlamos las frecuencias que se obtendrán, y mediante dilataciones controlaremos el espacio temporal que analizaremos.

Definimos pues el parámetro ‘a’ como parámetro de escala, el cuál determinará lo amplia que es la función que estamos utilizando, y el parámetro ‘b’ como el momento temporal el cual estamos analizando.

Antes de continuar, se exponen los puntos clave de lo que se ha explicado:

1. Cuanto más localizado en el tiempo esté el trozo de señal que observamos, es decir, cuanto más pequeño sea el parámetro  $a$ , más contenido de alta frecuencia estaremos obteniendo, ya que la longitud de la onda equivalente al trozo observado será más pequeña.
2. Por consiguiente, cuanto más grande sea el parámetro  $a$ , más contenido de baja frecuencia podremos recopilar, ya que éstos no están localizados en ningún punto en concreto, sino que están presentes en la totalidad de la señal.
3. Cuando el parámetro  $a$  tenga la longitud de la señal al completo, estaremos observando todos los componentes frecuenciales de la función, pero no sabremos dónde estarán actuando.
4. En el límite opuesto, cuando el parámetro  $a$  sea infinitesimal, tendremos información exacta del momento en el que nos encontramos, pero habremos descartado todo componente frecuencial existente.

Podríamos tener un dilema sobre qué tamaño de  $a$  utilizar y dónde, según la información que nos interese obtener, pero no es el caso. La transformada de wavelet obtiene toda la información de toda la señal haciendo un recorrido por toda ella variando además el valor de la escala. La idea es utilizar las escalas más grandes para extraer las bajas frecuencias y las pequeñas para extraer el contenido de alta frecuencia<sup>3</sup>. Al mismo tiempo, todo este proceso, cuando nos referimos a señales discretizadas, puede realizarse mediante un rápido y efectivo algoritmo, más rápido que el de la transformada de Fourier.

Las dilataciones y translaciones se hacen respecto a una wavelet de referencia, la cual se conoce como wavelet madre, y de la cual se obtendrá el subconjunto de wavelets que se utilizarán en la transformada. Esta wavelet madre tiene asociados los parámetros  $a=1$  y  $b=0$ .

La expresión de cualquier wavelet, en función de  $a$  y  $b$ , tiene la siguiente forma [3]<sup>4</sup>:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \cdot \psi\left(\frac{t-b}{a}\right); \quad a, b \in \mathbb{R}; \quad a \neq 0 \quad [3]$$

Y digo ‘cualquier wavelet’ porque hay muchos tipos de ellas. No es como hacer la transformada de Fourier. Diciendo transformada de wavelet no estamos dando toda la información que se necesita, ya que según cuales sean los propósitos, conociendo bien la señal, puede ser más útil una wavelet u otra. Algunos ejemplos son: Haar, Daubechies, Symlets, Shannon, etc. Se pueden observar algunas de ellas en la figura 3.2. Lo que sí que comparten todas las wavelets es, entre otras características, que su

integral respecto al tiempo es 0 (ondulan por encima y debajo del eje temporal) y que están bien localizadas.

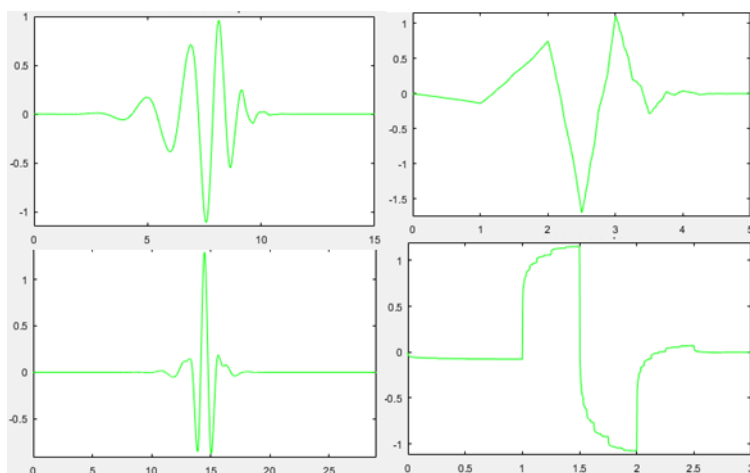


Figura 3.2. Diferentes tipos de wavelets (de izquierda a derecha y de arriba a abajo: db8, sym3, coif5, fk4) (Fuente: Wavelet display de Matlab)

La transformada de wavelet es un proceso reversible. El proceso de transformación se llama análisis y el proceso inverso para reconstruir la señal se llama síntesis. Las diferentes funciones wavelet están definidas, es decir que tienen una imagen concreta (en el caso de Haar oscila entre  $\pm 1$ ). Si queremos que se adapten a cualquier señal, sea cual sea la magnitud que tenga, se vuelve evidente que deben aparecer unos coeficientes que multipliquen esta función para adaptarse a la amplitud de la señal en cada momento. Estos coeficientes son lo que resulta cuando se hace la transformada de wavelet, y pueden ser de aproximación, que reflejan la tendencia general y suavizada de la señal, o de detalle, que reflejan los cambios bruscos y breves que contiene. En otras palabras, los coeficientes de aproximación retienen la baja frecuencia y los de detalle contienen la alta frecuencia.

Existen varios tipos de transformada de wavelet, pero nos centraremos en la DWT (transformada de wavelet discreta) y la CWT (transformada de wavelet continua).

La CWT es una herramienta que proporciona una representación de una señal variando la translación y la escala de una wavelet de forma continua. La transformada se expresa con la integral de la ecuación [4]. Ya que nos proporcionará un resultado gráfico escala-tiempo, utilizaremos esta transformada para ver donde se producen las frecuencias más elevadas.

$$CWT(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t) \bar{\Psi}\left(\frac{t-b}{a}\right) dt \quad [4]^{5,6}$$

Donde  $\bar{\Psi}$  es el conjugado de la wavelet madre,  $a$  es el parámetro de escala y  $b$  el parámetro de translación.

El parámetro de escalado se puede definir en función de la frecuencia<sup>7</sup> (ecuación 5, dónde  $f_c$ =frecuencia central de la wavelet), con lo que vemos que las más altas frecuencias se dan en las menores escalas. La frecuencia de la que se habla es una pseudo-frecuencia, condicionada por la frecuencia de muestreo que se ha utilizado en la recogida de datos. Hay que resaltar también que el componente frecuencial de cada escala, pese a tener un componente central, no será único, al no ser la wavelet madre una señal frecuencialmente pura como lo es una senoide.

$$a = \frac{f_c}{f} \quad [5]$$

La DWT es una transformada de wavelet en que las wavelets son discretizadas, y se utiliza una familia de wavelets ortonormales, en otras palabras, las wavelets son ortogonales y normalizadas para que su energía sea unitaria.

Para conseguir este subconjunto de wavelets, se le asocia al parámetro  $a$  y  $b$  unos valores específicos, en este caso potencias de 2 (ecuaciones 6 y 7).

$$a = 2^j \quad [6]$$

$$b = k \cdot 2^j \quad [7]$$

Donde  $j$  y  $k$  son enteros que dilatan y trasladan la wavelet madre respectivamente de forma que las diferentes wavelets sean ortonormales entre sí.

Si sustituimos estos nuevos parámetros en la ecuación 3, resulta la ecuación 8:

$$\Psi_{j,k}(t) = \frac{1}{\sqrt{2^j}} \cdot \Psi\left(\frac{t - k \cdot 2^j}{2^j}\right) \quad [8]$$

La señal quedará codificada mediante unos coeficientes de la siguiente forma<sup>2 5</sup>:

$$C_{jk} = \int_{-\infty}^{\infty} f(t) \cdot \frac{1}{\sqrt{2^j}} \cdot \Psi\left(\frac{t - k \cdot 2^j}{2^j}\right) dt \quad [9]$$

$$f(t) = C \cdot \Phi(t) + \sum_{j=0}^{n-1} \sum_k d_{jk} \cdot \Psi_{j,k}(t) \quad [10]$$

Donde  $f(t)$  es la señal original,  $C$  es el coeficiente de aproximación que multiplica a la función de escalado  $\Phi(t)$ ,  $d_j$  son los coeficientes de detalle a cada momento y  $\Psi_j, k(t)$  es la wavelet que se esté utilizando para el análisis.

Computacionalmente, se pueden interpretar los coeficientes como filtros pasa-alta frecuencia y pasa-baja frecuencia<sup>7</sup>. A medida que aumentemos el nivel de descomposición de la señal, con dilataciones y translaciones, los coeficientes le van otorgando un protagonismo a un tipo de frecuencias u otras.

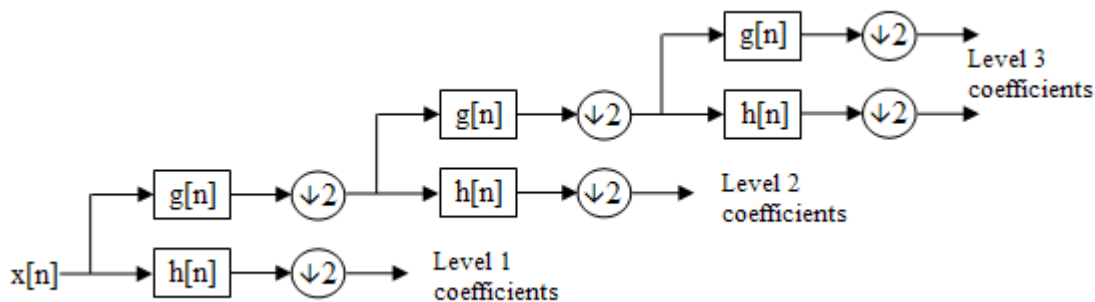


Figura 3.3. Proceso de descomposición de la DWT (Fuente: Wikipedia: Discrete Wavelet Transform)

Donde:

$S = a_3 + d_1 + d_2 + d_3$ , con  $S$ =señal,  $a_3$ =coeficientes de aproximación en el nivel 3, y  $d_i$ =coeficientes de detalle en el nivel  $i$ .

$G[n]$ =filtro bajas frecuencias

$H[n]$ =filtro altas frecuencias

Si nos fijamos en la figura 3.3, vemos cómo los coeficientes de aproximación resultan del filtro de baja frecuencia, y los coeficientes de detalle de los de alta. A cada nivel que avanza, vas obteniendo un contenido frecuencial distinto y con una localización temporal determinada, y lo que al final te queda es que tu señal es la suma de los coeficientes de aproximación finales con todos los detalles que se han ido quedando en el proceso. A medida que avanzamos de nivel, las frecuencias se vuelven más pequeñas (ya que nacen del filtro de bajas frecuencias anterior).

Dado al proceso de descomposición de la transformada de wavelet, la longitud de la señal de entrada tiene que ser múltiple de  $2^n$ <sup>8</sup>.

Podemos ver un ejemplo de cómo se descompone una señal, en forma de árbol, en la figura 3.4.



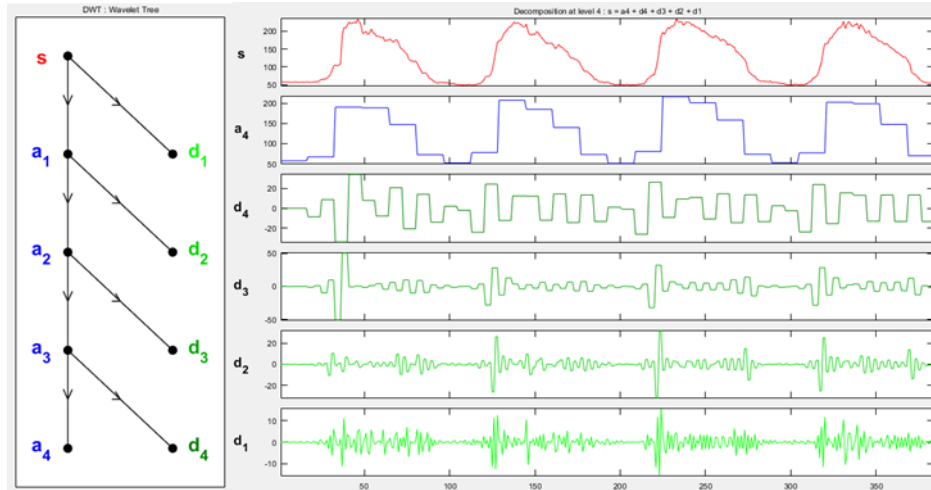


Figura 3.4. Descomposición de una señal  $S$  en 4 niveles mediante la wavelet de Haar (Fuente: Elaboración propia)

Usaremos esta herramienta para estudiar la evolución de la señal conforme se va aumentando el nivel de descomposición y para hacer el *denoising*, lo cual se explica más adelante.

### 3.2. Haar

La wavelet de Haar es la más antigua de todas, es la primera wavelet (1909). Es una función cuadrada que está definida como indica la ecuación 11:

$$\Psi(t) = \begin{cases} 1, & 0 \leq t < \frac{1}{2} \\ -1, & \frac{1}{2} \leq t < 1 \\ 0, & \text{otro} \end{cases} \quad [11]$$

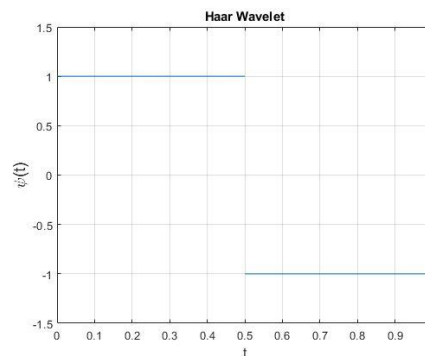


Figura 3.5. Wavelet de Haar (Fuente: Elaboración propia)

Su integral respecto al tiempo es 0 y su integral en valor absoluto vale 1 (ecuaciones 12 y 13).

$$\int_{-\infty}^{\infty} \Psi(t) dt = 0 \quad [12]$$

$$\int_{-\infty}^{\infty} |\Psi(t)| dt = 1 \quad [13]$$

Esta wavelet se encuentra bien localizada en el tiempo, entre 0 y 1 si hablamos de la wavelet madre. Podemos representar una señal como combinaciones de wavelets con distinta expansión y localización. En la figura 3.6 vemos un ejemplo de la wavelet de Haar con distintas escalas y localizaciones.

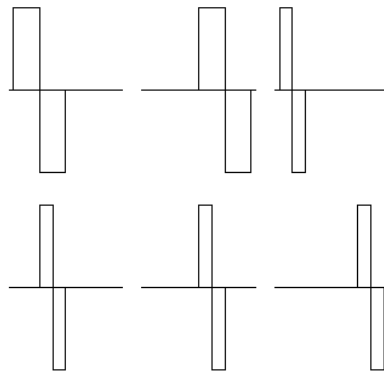


Figura 3.6. Dilataciones y translaciones de la wavelet de Haar en  $[0,1]$  (Fuente: Wavelets for kids)

Su frecuencia central,  $f_c$ , es de 1 Hz. Nos lo demuestra Matlab ejecutando la siguiente función (donde se le asignan 15 iteraciones):

```
>> centfrq('haar',15)
```

```
ans =
```

```
1.0000
```

Su simplicidad, la facilidad que otorga para interpretar resultados, y la facilidad de cálculo que ofrece, son algunas de las características tenidas en cuenta para decidir que es la wavelet que utilizaremos para el desarrollo de todo el trabajo.

### 3.3. Breve historia

Este capítulo es una pequeña presentación de las personas más influyentes con respecto a las wavelets y la cronología que se ha seguido. Son una herramienta muy reciente y no está descubierto todo el potencial de uso que pueden llegar a tener. Muchos de los nombres de las diferentes wavelets se deben a los nombres de estas personas.

La primera wavelet conocida se le atribuye a la wavelet de Haar<sup>9</sup> <sup>10</sup>, desarrollada por Alfred Haar en 1909. En 1946, Dennis Gabor presenta un trabajo muy similar a lo que conocemos por wavelet, señalando una señal en ventanas tiempo-frecuencia. En 1975 George Zweig desarrolla la transformada continua de wavelet, lo que sería complementado por la formulación de Pierre Goupillaud, Alexander Grossmann y Jean Morlet en 1982. Es muy destacable en el mundo de las wavelets el trabajo realizado por Ingrid Daubechies, quién en 1988 desarrolló el campo de las wavelets ortogonales compactas implementables por cortos filtros digitales.

La interpretación de la frecuencia en el tiempo de la CWT desarrollada por Nathalie Delprat también es muy importante.

Estas personas acabadas de nombrar son las que han establecido las bases del estudio de señales mediante wavelets y han contribuido a clarificar una parte del enorme potencial que poseen estas ondas.

### 3.4. Utilidad en el análisis de datos

Las funciones que han sido transformadas a series de Fourier están localizadas en el dominio frecuencial, pero no en el temporal. El inconveniente de este aspecto es que, si quieres hacer una pequeña modificación en el dominio frecuencial, sucederán cambios por toda la función en el dominio temporal, ya que las frecuencias no están normalmente localizadas en un único punto.

La gran ventaja de utilizar las wavelets es que están localizadas tanto en escala frecuencial (mediante dilataciones) y en la temporal (mediante translaciones), con lo cual podemos tener mucho más control sobre la función.

Otra ventaja que encontramos en las wavelets es que representan a las funciones de forma más compacta, es decir, para alcanzar una aproximación comparable normalmente se necesitan menos wavelets que las que se requiere utilizando senos y cosenos.

La wavelet de Haar es la más simple de todas. No es continua y por lo tanto no es derivable en todo su dominio. Sin embargo, esta propiedad lejos de ser una debilidad, la hace idónea para el análisis de

señales con transiciones repentinas, lo cual nos será útil a priori para el estudio del consumo eléctrico de los edificios cuando presenten picos de consumo que se alejan de la tendencia de la serie.

### 3.5. Presentación y explicación de las herramientas de Matlab que se van a utilizar.

Se va a utilizar además de los comandos de Matlab en la *command window* la herramienta waveletAnalyzer, que es una app de Matlab donde resulta más sencillo comenzar a conocer la señal.

#### 3.5.1. Estudio de la transformada mediante los display mode que ofrece Wavelet 1-D

##### 3.5.1.1. Show and scroll y Show and scroll (Stem coeffs.)

Con ello podemos ver la señal en un gráfico nivel-tiempo con una escala de colores donde nos indica la magnitud de los coeficientes en cada momento. Nos puede ayudar a ver de forma rápida si existe algún comportamiento frecuencial claro en nuestra señal. Para ilustrar esto ponemos el ejemplo de la señal 'noischir' que nos ofrece Matlab y hacemos una descomposición al nivel 6 con la wavelet de haar, (figura 3.7 y 3.8).

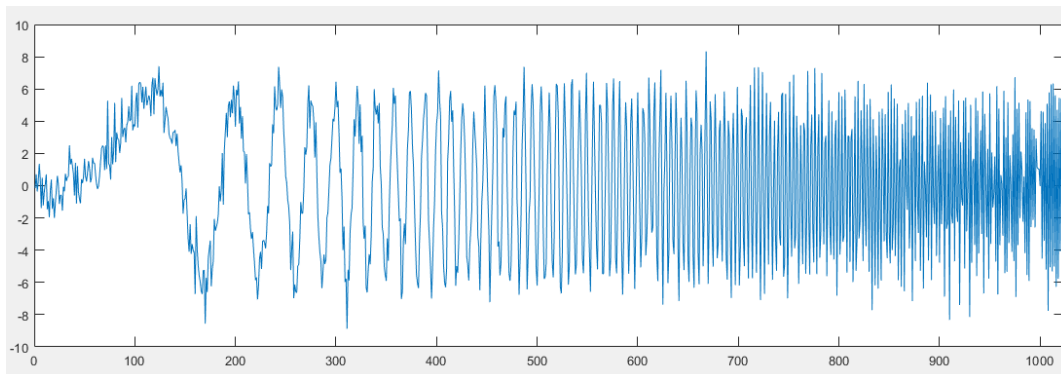


Figura 3.7. Señal 'noischir'. (Fuente: Matlab)

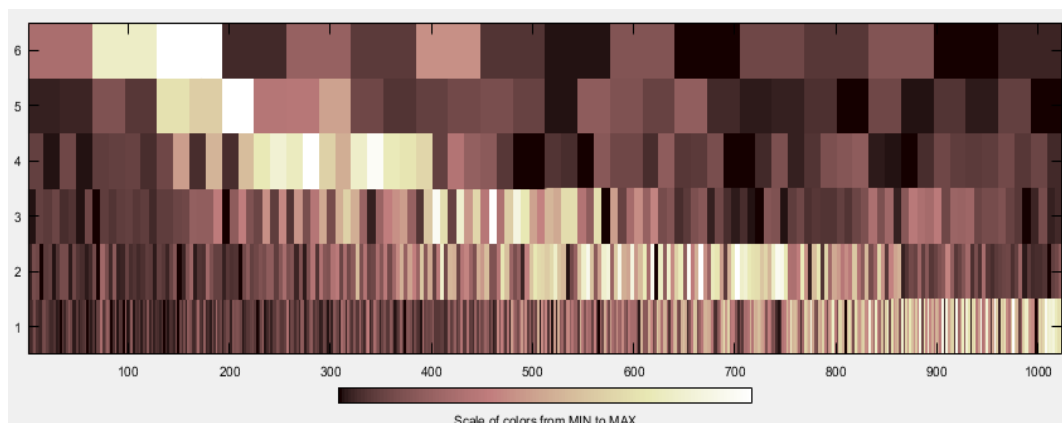


Figura 3.8. Coeficientes de los niveles 1-6 de la señal 'noischir' (Fuente: waveletAnalyzer de Matlab)

Vemos fácilmente como los coeficientes más altos (más característicos)<sup>8</sup> se van situando a niveles más bajos conforme pasa el tiempo, lo que se puede traducir en que la frecuencia de la señal aumenta a medida que pasa el tiempo.

### 3.5.1.2. Tree mode y Separate mode

Nos ayuda a observar la descomposición que se hace a cada nivel. Con ello podemos observar si los coeficientes son altos en según qué momento y a según qué nivel, dándonos información del contenido frecuencial que contiene cada periodo de tiempo (figura 3.9).

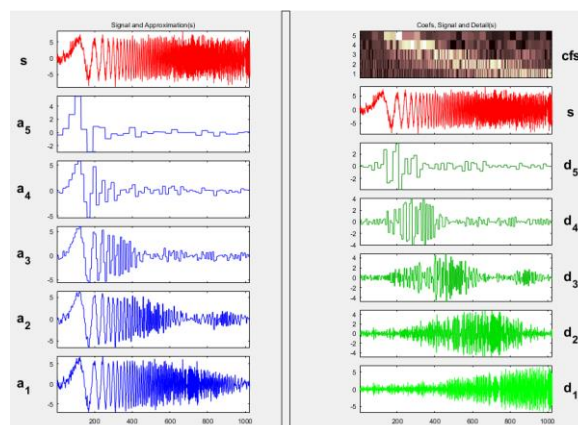


Figura 3.9. Descomposición en 5 niveles de la señal 'noischir' en 'Separate Mode' (Fuente: waveletAnalyzer de Matlab)

### 3.5.1.3. De-noise

Esta herramienta nos será de ayuda a la hora de eliminar aquellos componentes que contienen ruido y distorsionan el comportamiento natural de nuestra señal. Si somos capaces de identificar qué componentes son los que contienen el ruido, podremos eliminarlos y reconstruir la señal sin ellos.

A medida que se van filtrando los coeficientes con menor aportación al contenido de la señal, empezando generalmente por las frecuencias más elevadas, la misma va siendo más y más definida a nuestra vista, aunque no podemos descartar todos los componentes de alta frecuencia de nuestra señal sin perder gran cantidad de información por el camino<sup>3</sup>.

Es por ello que no usaremos un filtro global para toda la señal, sino que eliminaremos solo aquella porción de detalles que no excedan cierto límite. Esta técnica se conoce como *thresholding* y es mucho más eficiente. Así pues, para optimizar el filtrado, haremos un *thresholding* por niveles (*level-by-level*).

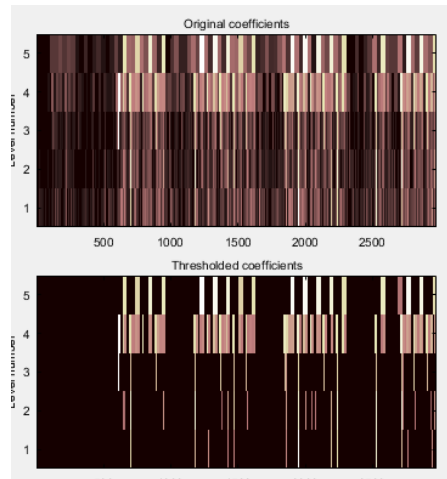


Figura 3.10. Ejemplo de una señal sin sus coeficientes más irrelevantes (Fuente: waveletAnalyzer de Matlab)

El thresholding de los coeficientes se aplica a los coeficientes de detalle, ya que son los más afectados por el ruido<sup>7</sup>. Los coeficientes de aproximación representan el comportamiento de baja frecuencia de la señal.

Existen dos técnicas para realizar el thresholding, que básicamente determinarán que se hará con los coeficientes más pequeños (darles valor nulo o encogerlos (*shrinkage*))<sup>7</sup>. Está el *hard thresholding* [14] y el *soft thresholding* [15].

$$\eta(x) = \begin{cases} x, & |x| \geq T \\ 0, & |x| < T \end{cases} \quad [14]$$

$$\eta(x) = \begin{cases} x - \lambda, & x > \lambda \\ 0, & |x| \leq \lambda \\ x + \lambda, & x < -\lambda \end{cases} \quad [15]$$

Donde  $\lambda$  es el valor del threshold y  $x$  es el vector de datos (en este caso son coeficientes de detalle).

*Soft thresholding* da mejores resultados cuando los coeficientes que se tratan contienen la información de la señal y del ruido.

Para determinar el valor de este límite (*threshold value*,  $\lambda$ ) hay diversos criterios. A continuación, se exponen los más relevantes<sup>7</sup>.

- Bayes

Este método se basa en asumir que las mediciones tienen “distribuciones de probabilidad a priori” (prior distribution) independientes dadas por un modelo de mezcla. Debido a que las mediciones se utilizan para estimar el peso en el modelo de mezcla, el método tiende a funcionar mejor con más muestras.

- BlockJS – Block James-Stein

Este método se basa en determinar el tamaño óptimo de un bloque y el threshold. El estimador de threshold del bloque resultante produce simultáneamente una adaptabilidad global y local óptima.

- FDR – False Discovery Rate

Este método se basa en el control de la proporción esperada de detecciones de falsos positivos de todas las detecciones positivas. El método FDR funciona mejor con datos dispersos. La elección de una relación,  $Q$ , menor que  $\frac{1}{2}$  produce un estimador *minimax* asintótico.

- Minimax

Este método utiliza un threshold fijo elegido para obtener un mínimo error cuadrático medio frente a un procedimiento ideal. El principio minimax se utiliza en estadística para diseñar estimadores.

- SURE

Escoge el valor del threshold basándose en la estimación imparcial del riesgo de Stein (SURE=Stein's Unbiased Estimate of Risk), la cual es una función de pérdida cuadrática. Obtiene una estimación del valor del threshold que minimiza el riesgo.

- Universal threshold

Utiliza un threshold fijo. Utiliza la siguiente ecuación para determinar el valor del threshold:

$$\lambda = \sqrt{2 \ln(N)} \quad [16]$$

Donde N es la longitud de la señal. Se puede utilizar este parámetro para toda la señal o se puede obtener un valor de threshold por cada nivel.

En este estudio, se aplicará el método que mejor resultado proporcione en el momento de hacerlo. Existen además dos maneras de determinar el ruido:

- Nivel independiente
  - o Calcula la varianza del ruido en función de los coeficientes de wavelets de la mejor escala (resolución más alta).
- Nivel dependiente
  - o Estima la varianza del ruido en función de los coeficientes de wavelet en cada nivel de resolución.

Aplicaremos el nivel dependiente, ya que queremos hacer un *thresholding level-by-level*.

Podemos realizar el *thresholding* mediante la App *waveletAnalyzer*. Para ello, descompondremos la señal y aplicaremos el método a cada uno de los coeficientes de detalle de cada nivel. Después recompondremos la señal.

### 3.5.2. Wavelet Coefficients Selection 1-D

Nos ayuda a ver cómo queda la señal a medida que añadimos o quitamos coeficientes determinados y así podemos determinar su importancia. Podemos elegir estos coeficientes de forma manual. La figura 3.11 muestra cómo se consigue una señal bastante fiel a la original utilizando una cuarta parte de sus coeficientes.



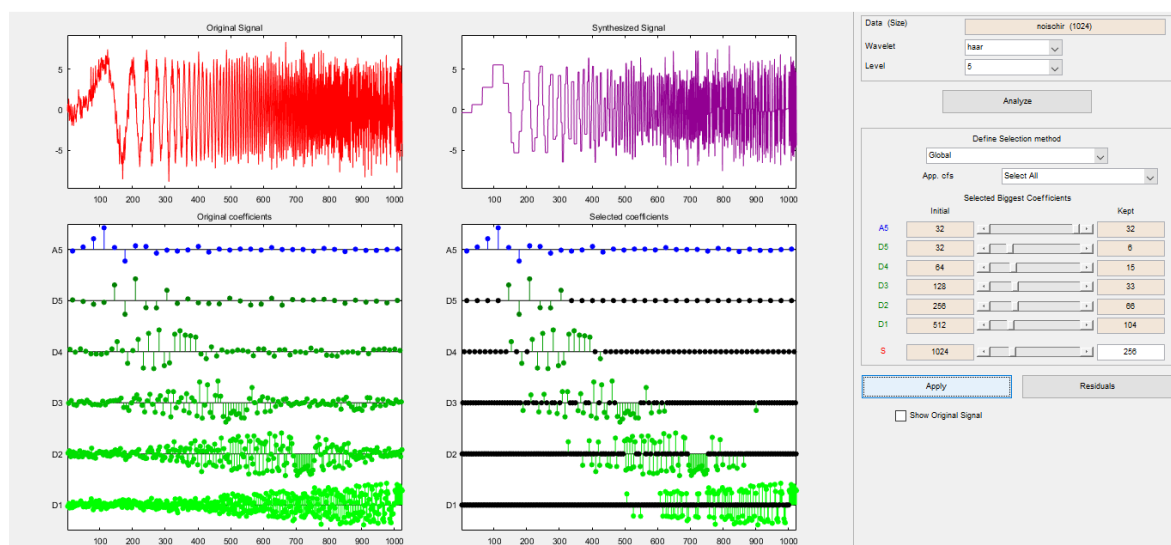


Figura 3.11. Selección de 256 coeficientes de los 1024 totales (Fuente: waveletAnalyzer de Matlab)

### 3.5.3. Continuous Wavelet 1-D

Utilizando la herramienta *Continuous Wavelet 1-D*, podremos analizar los conos de influencia que están presentes. Podemos ver cómo son los coeficientes a diferentes escalas, y nos muestra las 'líneas máximas', que nos enseñan a qué escala aparecen los máximos coeficientes en cada momento.

Las líneas máximas son marcas que indican los momentos en que se producen las máximas modificaciones en el patrón de comportamiento de la señal, en función de los componentes frecuenciales asociados tanto a la wavelet como a la señal. Vienen a representar el espectro frecuencial de la perturbación detectada usando todas las escalas posibles.

A mayor escala, menor es la frecuencia, y los conos de influencia lo que nos muestra es que en la escala más baja se encuentra el vértice, es decir que está bien localizado en el tiempo, y a medida que aumentan las escalas se va expandiendo y deslocalizando el componente frecuencial.

Además, se utilizará esta herramienta para que nos marque la escala máxima que hay que utilizar para analizar una señal de una longitud determinada para que el resultado no se vea alterado.

### 3.5.4. Wavelets packets 1-D

Los 'paquetes de wavelet' son herramientas en las que la descomposición por niveles se aplica tanto a los coeficientes de aproximación como a los coeficientes de detalle (ver figura 3.12), y no solo a los de aproximación como pasa con la DWT. Los utilizaremos como otra herramienta de análisis adicional, aunque en la pestaña de *wavelet packets 1-D* de la App *waveletAnalyzer* se puede poner el modo 'wavelet tree' si se prefiere.

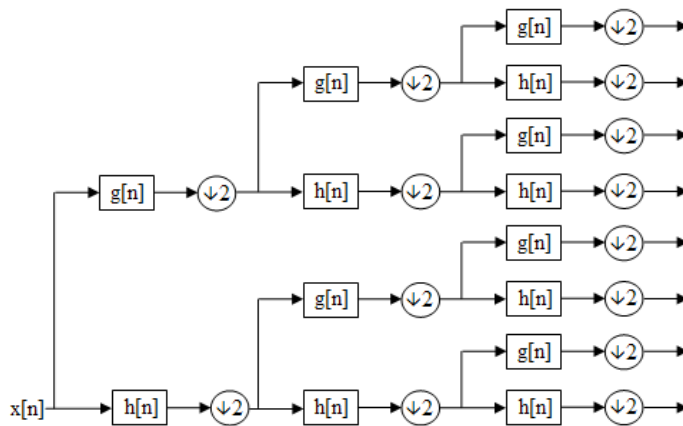


Figura 3.12. Descomposición de una señal usando un paquete wavelet (Fuente: waveletAnalyzer de Matlab)

En la figura 3.13 tenemos un ejemplo con el gráfico frecuencia-tiempo (en colores azul y magenta), donde podemos ver fijándonos en el ancho de las bandas de colores que, por niveles, cuanto más resolución temporal hay peor es la resolución frecuencial. Lo que se ve en el dibujo es el árbol en sí. La banda que está más arriba tiene la misma altura que todas las demás, la segunda tiene la altura de la suma de las dos de abajo, y las dos últimas son los detalles y aproximación del último nivel. (En orden desde arriba hacia abajo: nodo (1,1), nodo (2,1), nodo (3,1) y nodo (3,0)).

Aun así, lo que nos indican los colores de los coeficientes es dónde hay más variedad frecuencial. (por ejemplo, vemos como a las 7:30h de la mañana (valor 30) hay mucho contenido frecuencial (tanto alto como bajo), y en cambio hay otras zonas donde predomina la baja frecuencia. Las zonas con alto contenido frecuencial son las más sensibles a contener ruido.

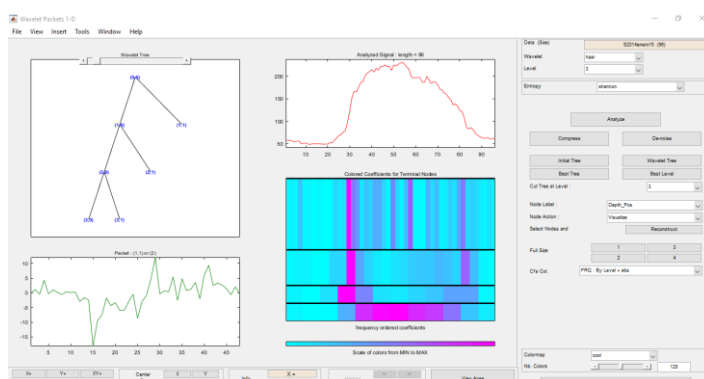


Figura 3.13. Wavelet Packets 1-D con la señal del 15 de enero de 2014 (Fuente: waveletAnalyzer de Matlab)

Vemos como los máximos coeficientes de aproximación se dan en la parte central de la señal, donde se encuentra el máximo valor de consumo y hace la señal casi simétrica a ambos lados, y los máximos coeficientes de detalle suceden en puntos de cambio bruscos de la señal.

## 4. Exposición de los datos y presentación de las características de los edificios a estudiar.

En este documento se analizan, como ejemplo para avanzar en el desarrollo y depuración de una metodología de análisis y predicción, los consumos eléctricos anuales de un centro de la *Universitat Politècnica de Catalunya*, recogidos en intervalos de 15 minutos, de los años 2014, 2015, 2016 y 2017.

La escuela es ETSEIB (*Escola Tècnica Superior d'Enginyeria Industrial de Barcelona*), y se encuentra en la avenida diagonal de Barcelona, nº 647. Tiene una superficie total de 47.000 m<sup>2</sup>, distribuidos en sus 12 plantas y, entre trabajadores y alumnos, se concentran alrededor de 3200 personas.

A continuación, se exponen los calendarios laborales de Barcelona de los años comentados<sup>11</sup>, y además los festivos propios de la universidad<sup>12</sup>. Todo ello es necesario para conocer más la señal a analizar.

### 4.1. Calendarios laborales de Barcelona (2014-2017)

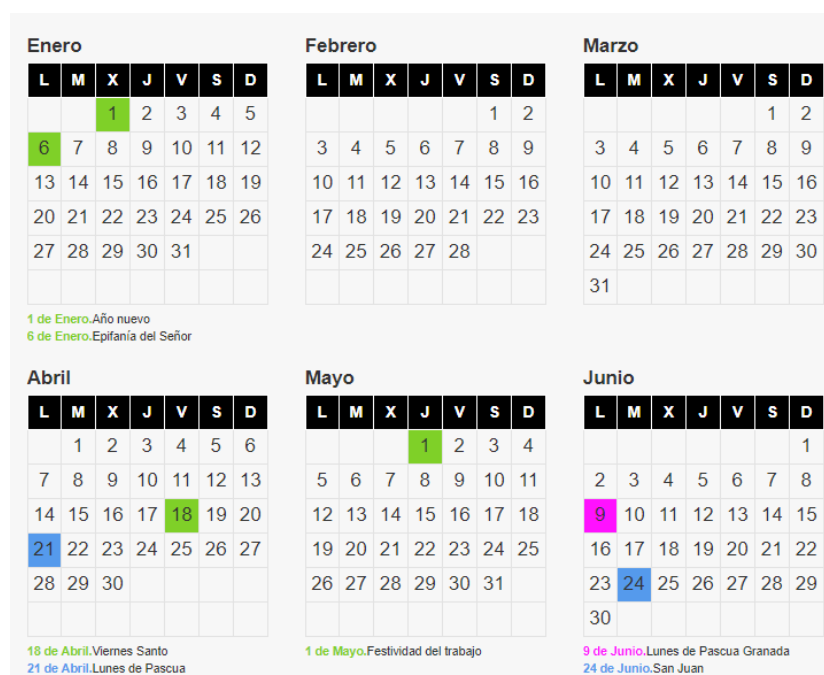


Figura 4.1. Primer semestre del calendario laboral de Barcelona en 2014 (Fuente: <https://www.calendarioslaborales.com/>)

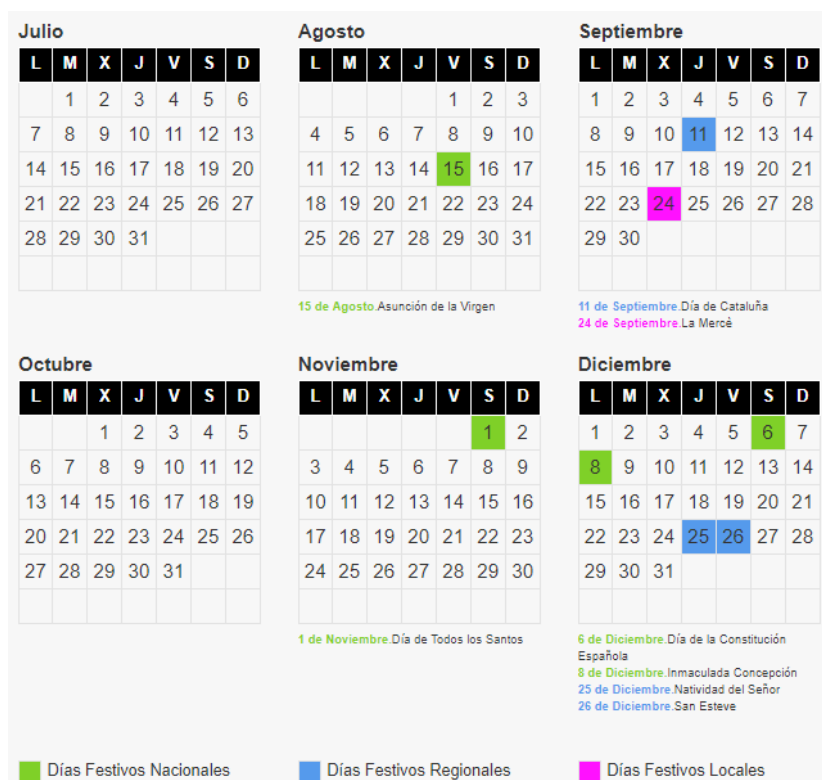


Figura 4.2. Segundo semestre del calendario laboral de Barcelona en 2014 (Fuente: <https://www.calendarioslaborales.com/>)

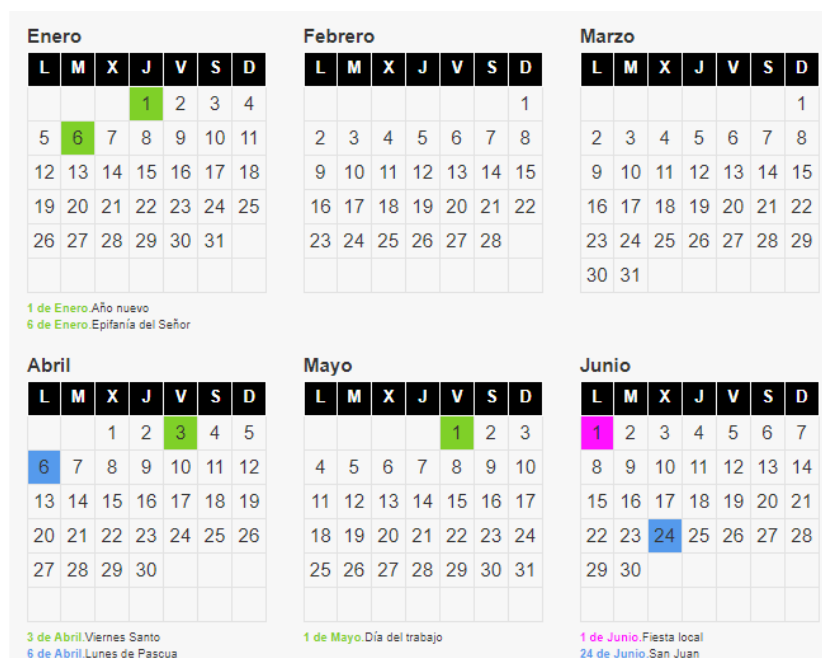


Figura 4.3. Primer semestre del calendario laboral de Barcelona en 2015 (Fuente: <https://www.calendarioslaborales.com/>)

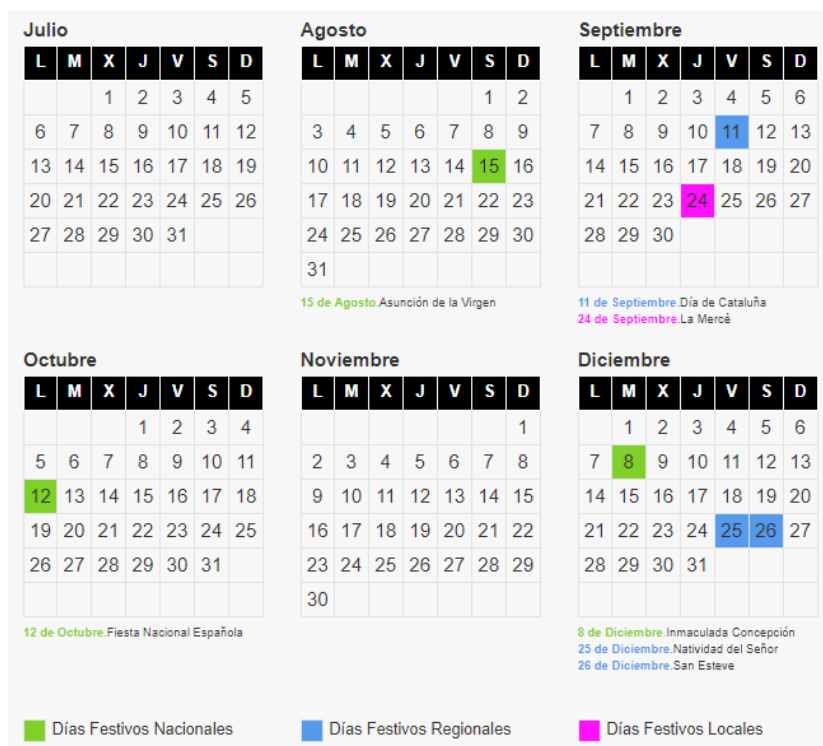


Figura 4.4. Segundo semestre del calendario laboral de Barcelona en 2015 (Fuente: <https://www.calendarioslaborales.com/>)

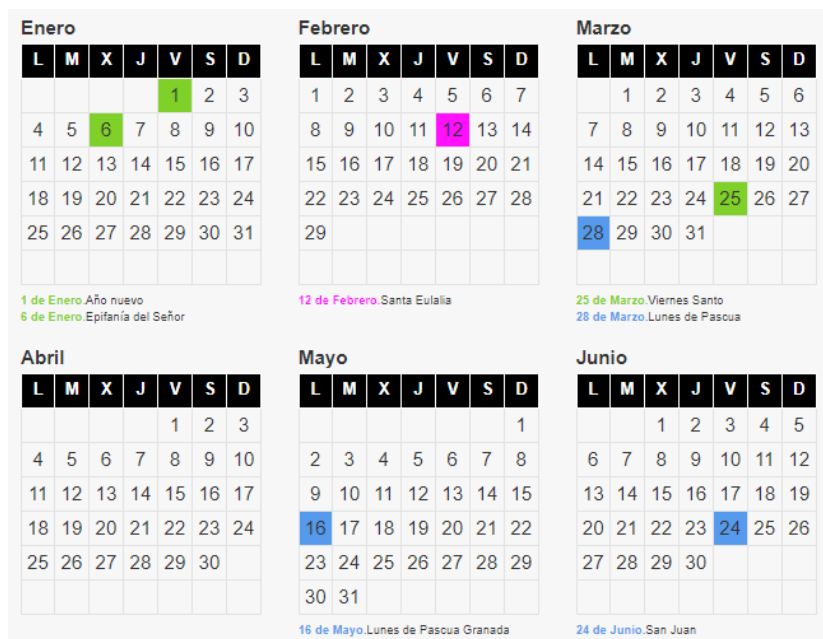


Figura 4.5. Primer semestre del calendario laboral de Barcelona en 2016 (Fuente: <https://www.calendarioslaborales.com/>)

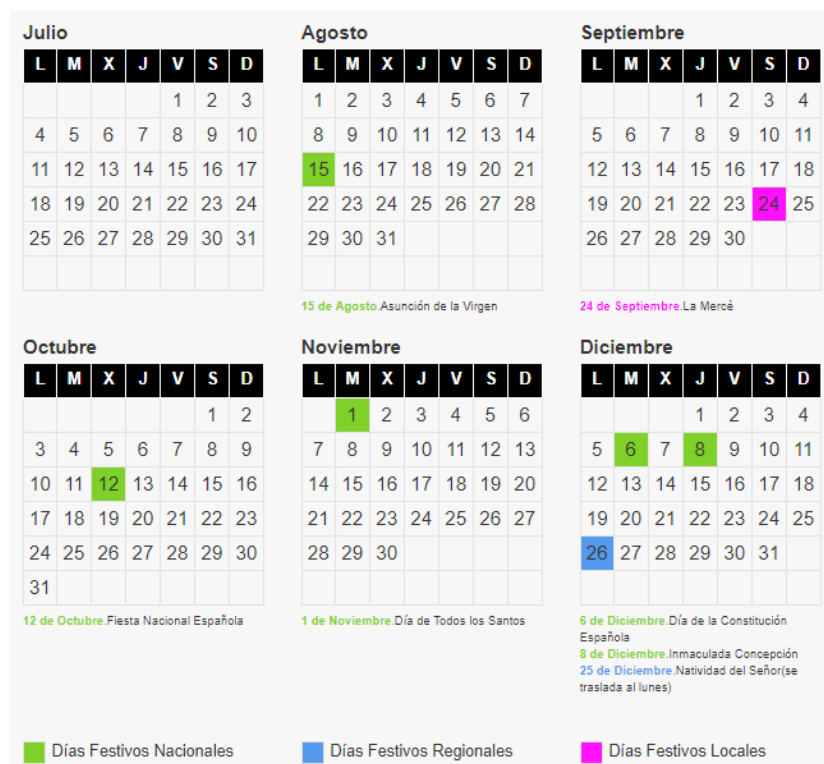


Figura 4.6. Segundo semestre del calendario laboral de Barcelona en 2016 (Fuente: <https://www.calendarioslaborales.com/>)

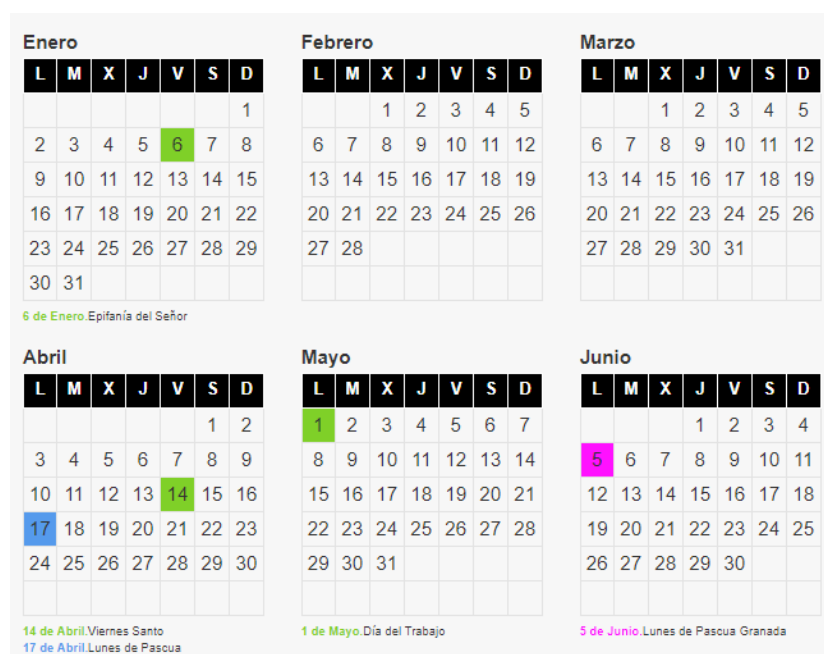


Figura 4.7. Primer semestre del calendario laboral de Barcelona en 2017 (Fuente: <https://www.calendarioslaborales.com/>)

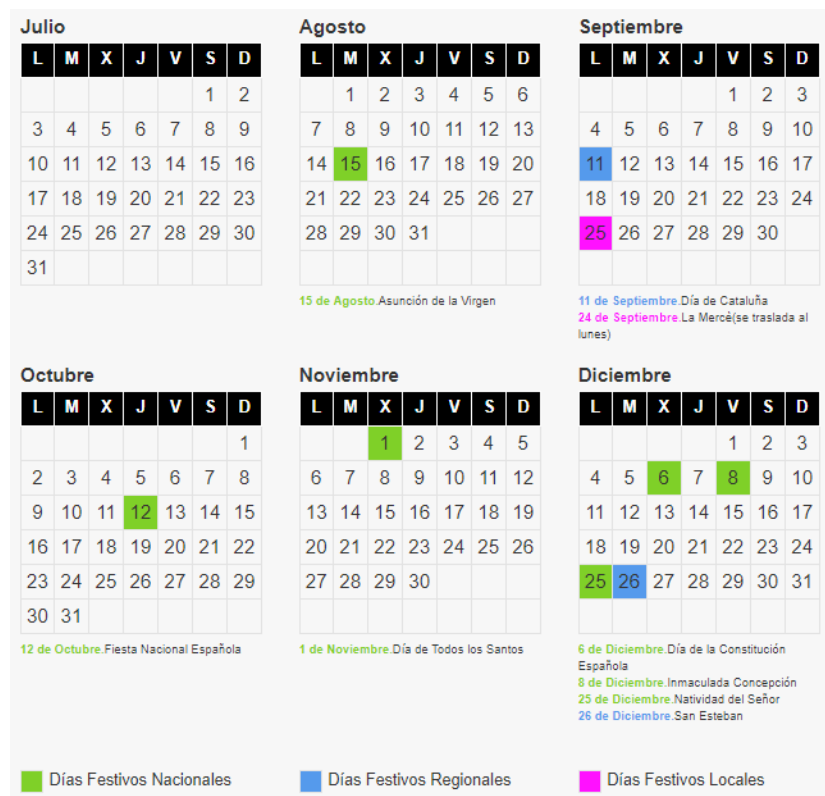


Figura 4.8. Segundo semestre del calendario laboral de Barcelona en 2017 (Fuente: <https://www.calendarioslaborales.com/>)

#### 4.1.1. Festivos propios de la universidad

2014	2015	2016	2017
1 de enero	1 de enero	1 de enero	6 de enero
6 de enero	6 de enero	6 de enero	14 de abril
1 de mayo	3 de abril	25 de marzo	17 de abril
24 de junio	6 de abril	28 de marzo	1 de mayo
11 de septiembre	1 de mayo	16 de mayo	24 de junio
1 de noviembre	24 de junio	24 de junio	15 de agosto
6 de diciembre	15 de agosto	15 de agosto	11 de septiembre
8 de diciembre	11 de septiembre	12 de octubre	12 de octubre
25 de diciembre	12 de octubre	1 de noviembre	1 de noviembre
26 de diciembre	8 de diciembre	6 de diciembre	6 de diciembre
	25 de diciembre	8 de diciembre	8 de diciembre
	26 de diciembre	26 de diciembre	25 de diciembre
			26 de diciembre

Figura 4.9. Festivos generales ETSEIB 2014-2017 (Fuente: <https://www.upc.edu/sga/es/normativas/CalendariosAcademicos/>)

## 4.2. Datos recogidos

Los datos de consumo durante los años 2014, 2015, 2016 y 2017 fueron recogidos cada 15 minutos. A continuación, se presentan los consumos de cada uno de estos años gráficamente con sus características básicas en forma de tabla.

### 4.2.1. Año 2014

Tabla 4.1. Consumo ETSEIB 2014 (kWh)

Consumo ETSEIB 2014 [kWh]	
<b>TOTAL</b>	3.078.363,01
<b>PROMEDIO</b>	87,85
<b>MEDIANA</b>	59,00
<b>MAX</b>	282,63
<b>MIN</b>	0,00

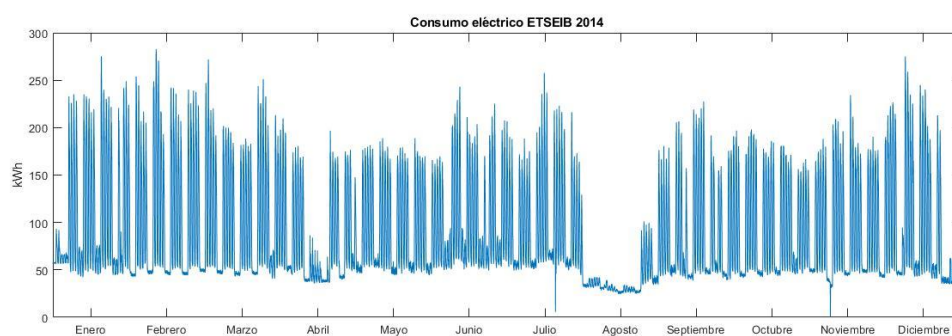


Figura 4.10. Consumo eléctrico ETSEIB 2014 en kWh

### 4.2.2. Año 2015

Tabla 4.2. Consumo ETSEIB 2015 (kWh)

Consumo ETSEIB 2015 [kWh]	
<b>TOTAL</b>	3.319.353,44
<b>PROMEDIO</b>	96,62
<b>MEDIANA</b>	56,86
<b>MAX</b>	239.806,03
<b>MIN</b>	0,00



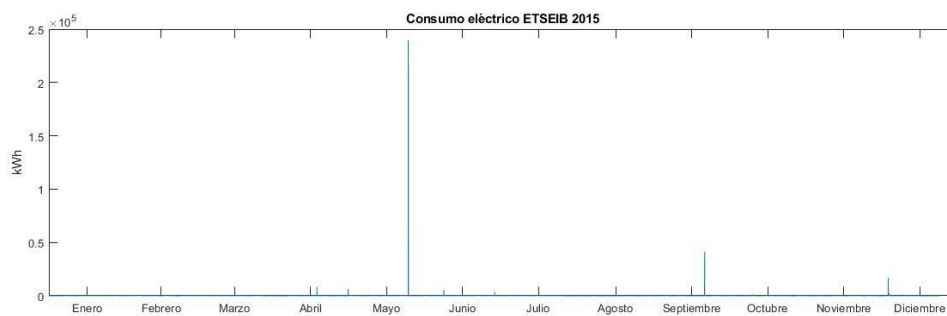


Figura 4.11. Consumo eléctrico ETSEIB 2015 en kWh

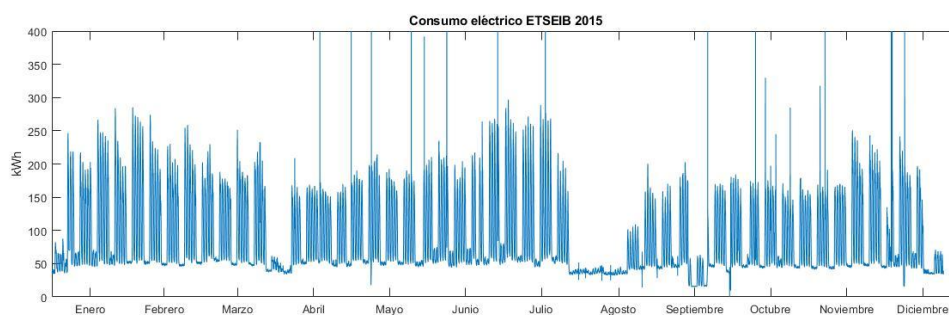


Figura 4.12. Consumo eléctrico ETSEIB 2015 en kWh (aumentado)

### 4.2.3. Año 2016

Tabla 4.3. Consumo ETSEIB 2016 (kWh)

Consumo ETSEIB 2016 [kWh]	
<b>TOTAL</b>	2.886.289,75
<b>PROMEDIO</b>	82,21
<b>MEDIANA</b>	54,19
<b>MAX</b>	2.593,15
<b>MIN</b>	11,86

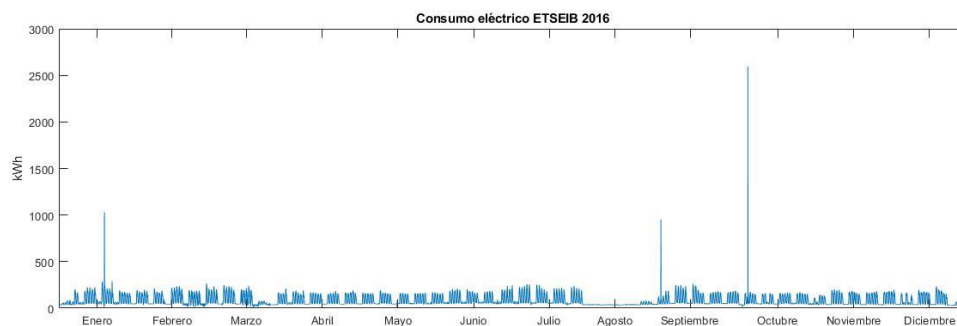


Figura 4.13. Consumo eléctrico ETSEIB 2016 en kWh

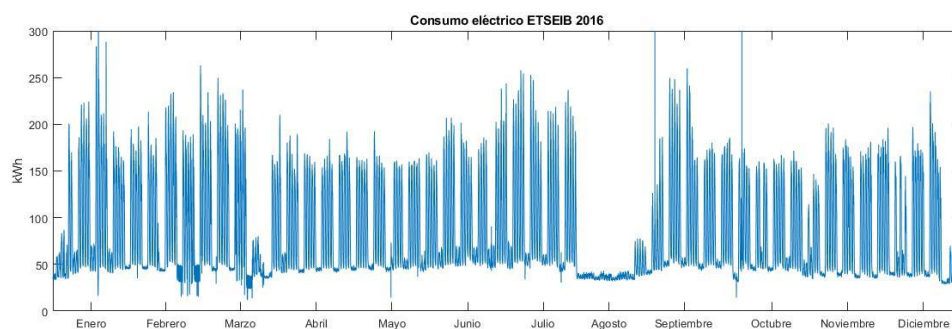


Figura 4.14. Consumo eléctrico ETSEIB 2016 en kWh (aumentado)

#### 4.2.4. Año 2017

Tabla 4.4. Consumo ETSEIB 2017 (kWh)

Consumo ETSEIB 2017 [kWh]	
<b>TOTAL</b>	2.373.538,65
<b>PROMEDIO</b>	68,59
<b>MEDIANA</b>	40,83
<b>MAX</b>	24.857,32
<b>MIN</b>	9,05

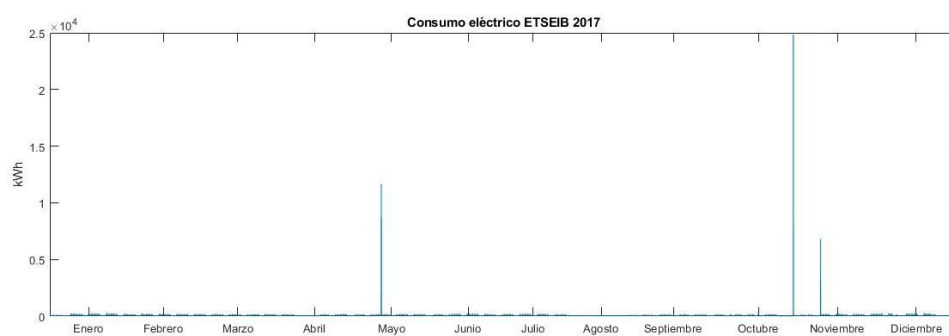


Figura 4.15. Consumo eléctrico ETSEIB 2017 en kWh

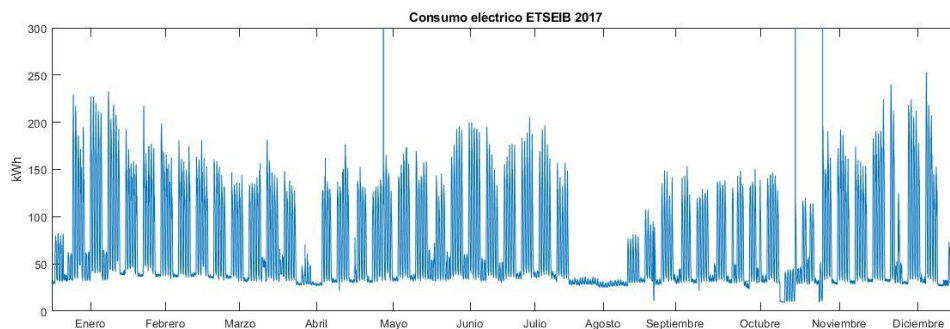


Figura 4.16. Consumo eléctrico ETSEIB 2017 en kWh (aumentado)

A simple vista, lo que llama la atención es que en todos los años excepto en el 2014 podemos observar unos picos de consumo elevadísimos en diferentes momentos del año cada vez. La rapidez en la que se producen, el consumo altísimo al que se llega y la falta de periodicidad en comparación con otros años nos indica que estos consumos son producto de anomalías en el normal comportamiento de la señal. Pueden ser debidos a fallos en la instalación eléctrica o a fallos en los aparatos de medida, pero, en cualquier caso, no forman parte de la normalidad y la lógica de los datos, con lo cual cuando se limpie la señal de ruido claramente estos picos desaparecerán.

## 5. Análisis frecuencia-tiempo de los datos

### 5.1. Explicación previa

En este capítulo estudiaremos los datos de 2014. Se podrá utilizar el método para los otros años cambiando las señales iniciales a utilizar. Como apunte, cuando se proceda a analizar los años 2015, 2016 y 2017 lo primero que se tendrá que hacer es cambiar los picos de consumo que hemos visto en el apartado 4.2 de los años 2015, 2016 y 2017 por valores acordes a su entorno. Es evidente que son anomalías, y por ello hay que eliminarlas para que no contamine el estudio de las señales.

La estrategia que se seguirá es la siguiente:

1. Realizar la descomposición en 2 niveles de la señal de los días a estudiar<sup>13</sup>.
  - a. Los días escogidos son:
    - i. Miércoles 15 de enero de 2014 (lectivo, en periodo de exámenes). Señal: S2014enero15.
    - ii. Miércoles 9 de abril de 2014 (lectivo, en periodo de clases). Señal: S2014abril9.
    - iii. Miércoles 13 de agosto de 2014 (festivo, verano). Señal: S2014agosto13.
    - iv. Miércoles 31 de diciembre de 2014 (festivo, navidad). Señal: S2014diciembre31.
2. Realizar el de-noising de la señal original de los días a estudiar<sup>14</sup> para eliminar las partes de la señal que no aporten información.
  - a. El método de thresholding escogido es *Heuristic SURE*<sup>15,16</sup>.
3. Realizar la descomposición en 2 niveles de las señales obtenidas en el punto 2.
4. Realizar la transformada continua de wavelet (CWT) a la señal sin descomponer<sup>17</sup>.
5. Realizar la transformada continua de wavelet (CWT) a los coeficientes de aproximación y de detalle resultantes de la descomposición.

Se realiza el thresholding para limpiar la señal previamente, para hacer un primer filtrado de 'ruido'. Después analizaremos lo que nos quede de la señal. No tiene por qué tratarse de ruido en el sentido de interferencias en la toma de datos, sino que se puede definir como ondas no predecibles.

Con la transformada discreta de wavelet (DWT) obtendremos unos coeficientes distintos en cada nivel, con lo que separaremos los componentes de la señal según tengan comportamiento frecuencial alto o bajo. Una vez tengamos separados estos coeficientes, analizaremos con la transformada continua de wavelet (CWT) cada uno de ellos en un gráfico escala-tiempo.

Utilizaremos los coeficientes de aproximación para conocer la tendencia general que sigue la serie de datos, y los coeficientes de detalle para localizar picos de consumo rápidos y, dependiendo a qué nivel de detalle estén localizados, intentar identificar a qué son debidos.

## 5.2. Procedimiento de análisis

El consumo, en kWh, de los días escogidos para estudiar se muestra en la figura 5.1. Los valores que aparecen en el eje de abscisas es el número de datos que tiene una señal diaria, con inicio a las 00:00h y final a las 23:45. En este caso va del 1 al 96 ya que se tiene 1 valor cada 15 minutos.

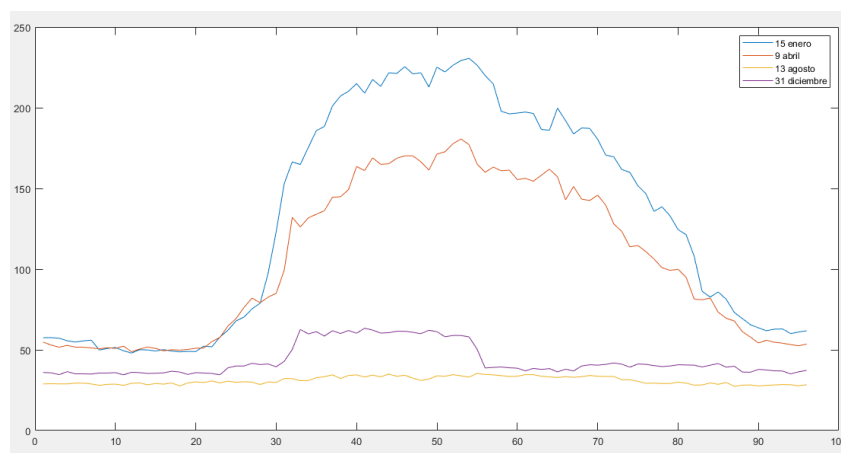


Figura 5.1. Consumo (kWh) diario ETSEIB 2014

En enero el consumo es mayor que en abril, pero guardan similitud en su comportamiento. El 31 de diciembre vemos cómo de 08:00h a 14:00h funcionan los servicios mínimos y en agosto no hay actividad, pero sí hay un consumo mínimo constante.

Descomponemos cada señal en 2 niveles por su corta extensión y procedemos a realizar el de-noising por niveles.

### 5.2.1. Descomposición de las señales originales

Utilizaremos primeramente la app WaveletAnalyzer en su modo *Wavelet 1-D*, y usando la wavelet de Haar y dos niveles de descomposición obtenemos lo siguiente:

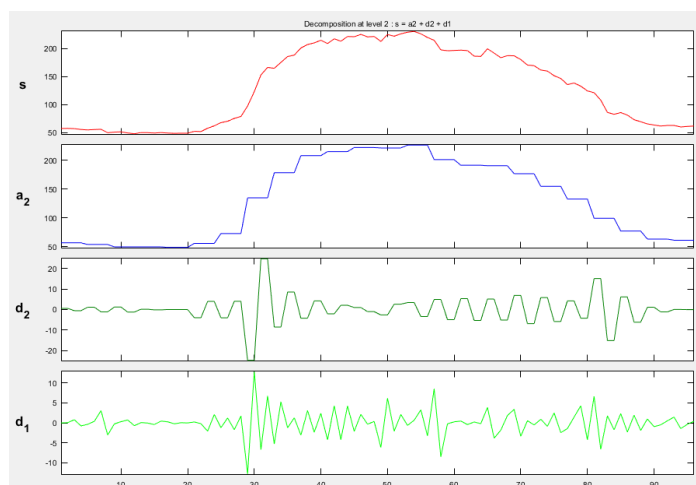


Figura 5.2. Descomposición con Haar de la señal del 15 de enero de 2014 en dos niveles

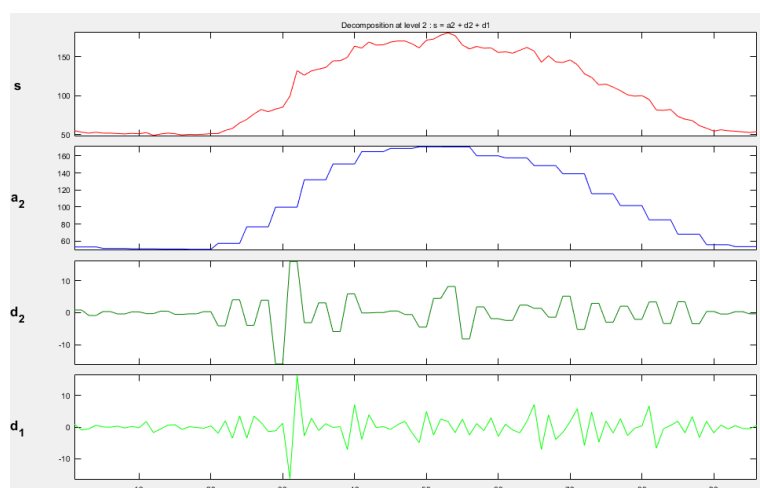


Figura 5.3. Descomposición con Haar de la señal del 9 de abril de 2014 en dos niveles

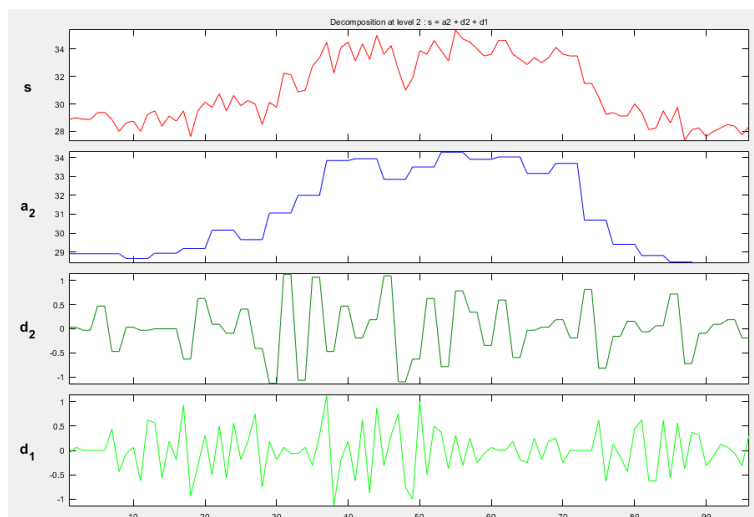


Figura 5.4. Descomposición con Haar de la señal del 13 de agosto de 2014 en dos niveles

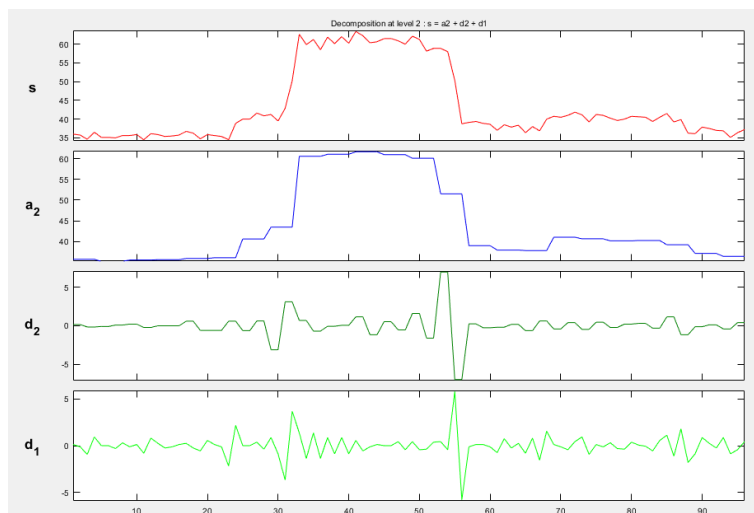


Figura 5.5. Descomposición con Haar de la señal del 31 de diciembre de 2014 en dos niveles

Ahora aplicamos el *thresholding* usando también la interfaz que nos proporciona esta app, en la pestaña *De-noise*.

## 5.2.2. De-noising de las señales originales

Select thresholding method

Heuristic SURE

☒ soft ☐ hard

Select noise structure

Unscaled white noise

Lev	Int	Select	Thresh
2	1		0.062
1	1		0.619

Figura 5.6. Thresholding aplicado a la señal del 15 de enero de 2014 (HS)

Select thresholding method

Heuristic SURE

☒ soft ☐ hard

Select noise structure

Unscaled white noise

Lev	Int	Select	Thresh
2	1		0.094
1	1		0.221

Figura 5.7. Thresholding aplicado a la señal del 9 de abril de 2014 (HS)

Select thresholding method

Heuristic SURE

☒ soft ☐ hard

Select noise structure

Unscaled white noise

Lev	Int	Select	Thresh
2	1		2.250
1	1		1.591

Figura 5.8. Thresholding aplicado a la señal del 13 de agosto de 2014 (HS)

Vemos que el algoritmo de Heuristic SURE elimina todos los coeficientes de detalle de la señal de agosto, debido probablemente a su bajo valor. Para no alterar la señal en demasía, retocaremos el thresholding de modo que los residuales de la señal queden en el entorno de los demás.



☒ soft    ☐ hard  
 Select noise structure  
 Unscaled white noise  

Lev	Int	Select	Thresh
2	1		0.293
1	1		0.207

Figura 5.9. Thresholding aplicado a la señal del 13 de agosto de 2014 (manual)

Select thresholding method  
 Heuristic SURE  
☒ soft    ☐ hard  
 Select noise structure  
 Unscaled white noise  

Lev	Int	Select	Thresh
2	1		1.375
1	1		1.326

Figura 5.10. Thresholding aplicado a la señal del 31 de diciembre de 2014 (HS)

De nuevo, el método de determinación de umbral considera los coeficientes de detalle como ruido, debido a los niveles de análisis y a la estructura de la señal, por lo que al realizarles el thresholding automático no hallamos resultados satisfactorios. Para que el método sea menos restrictivo, lo modificaremos manualmente.

☒ soft    ☐ hard  
 Select noise structure  
 Unscaled white noise  

Lev	Int	Select	Thresh
2	1		0.681
1	1		0.422

Figura 5.11. Thresholding aplicado a la señal del 31 de diciembre de 2014 (manual)

Las señales después de realizar el de-noising han dejado los siguientes residuales:

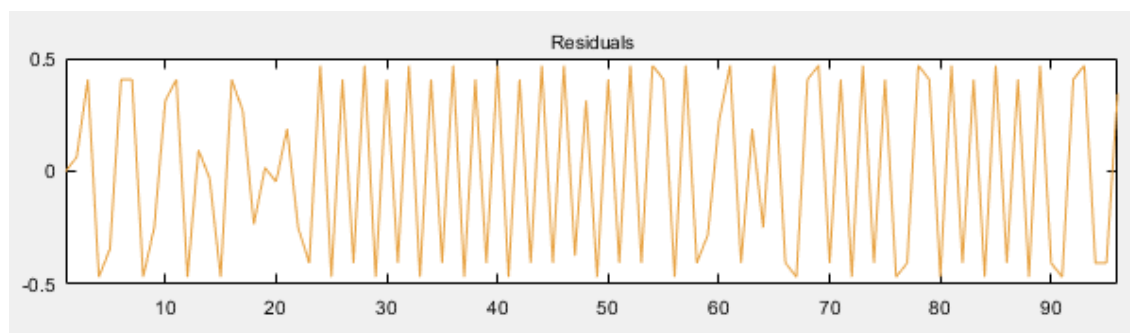


Figura 5.12. Residuales del thresholding aplicado a la señal del 15 de enero de 2014 (HS)

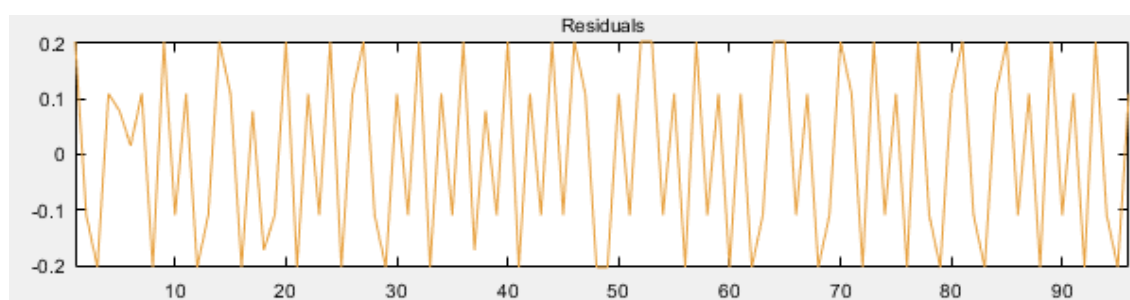


Figura 5.13. Residuales del thresholding aplicado a la señal del 9 de abril de 2014 (HS)

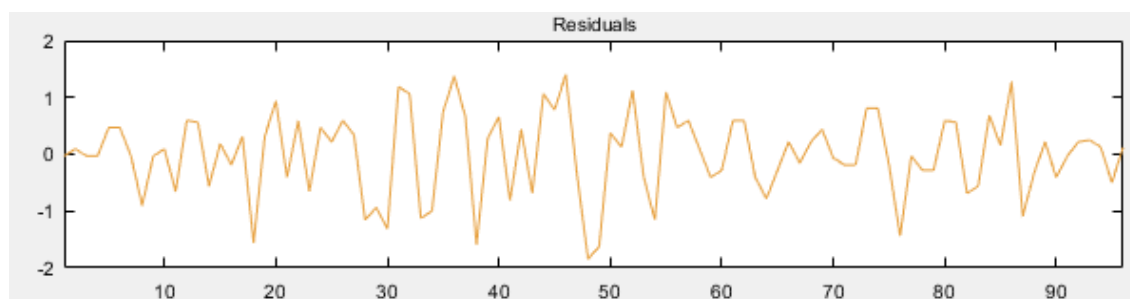


Figura 5.14. Residuales del thresholding aplicado a la señal del 13 de agosto de 2014 (HS)

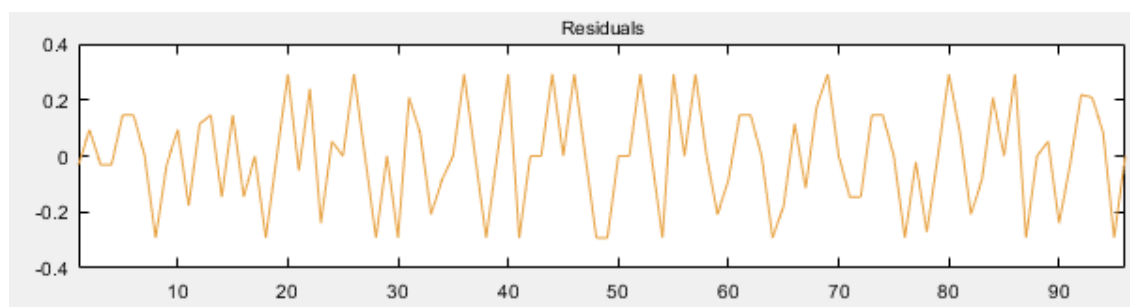


Figura 5.15. Residuales del thresholding aplicado a la señal del 13 de agosto de 2014 (manual)

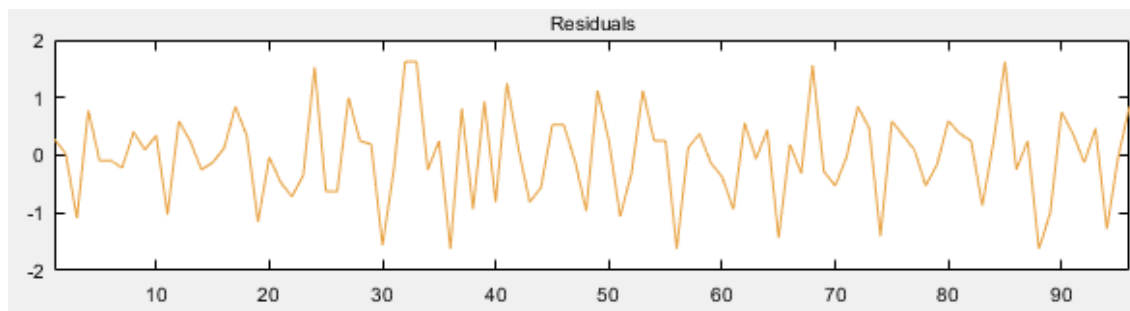


Figura 5.16. Residuales del thresholding aplicado a la señal del 31 de diciembre de 2014 (HS)

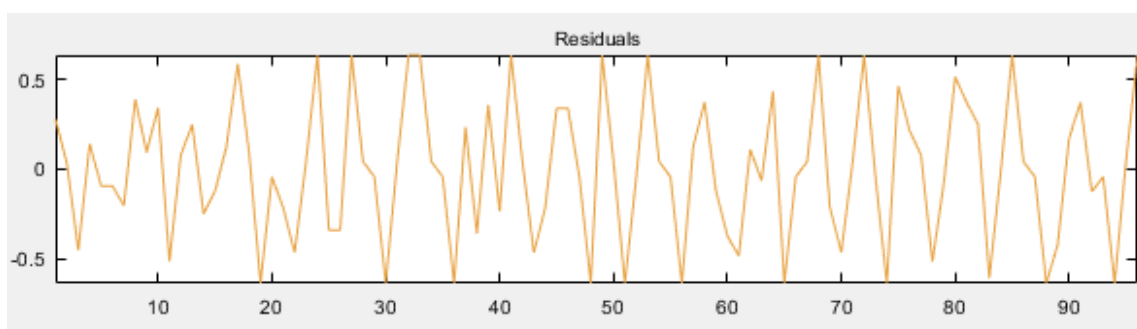


Figura 5.17. Residuales del thresholding aplicado a la señal del 31 de diciembre de 2014 (manual)

Lo que se busca con el *de-noising* no es variar la señal hasta tal punto de perder información valiosa, sino eliminar estos residuales que como vemos están presentes en toda la señal con una forma de onda que relacionamos con oscilaciones de la señal no predecibles o ruido.

Guardamos las nuevas señales con una distintiva 'd' al final para saber que se ha realizado el de-noising (por ejemplo, S2014enero15 pasa a ser S2014enero15d), y las cargamos en el workspace.

### 5.2.3. Descomposición de las señales sin ruido

Haremos lo mismo que en el capítulo 5.2.1. pero con las nuevas señales obtenidas en el capítulo 5.2.2. y utilizando esta vez los comandos de Matlab en la *command window*. Los comandos están en el anexo A.1.

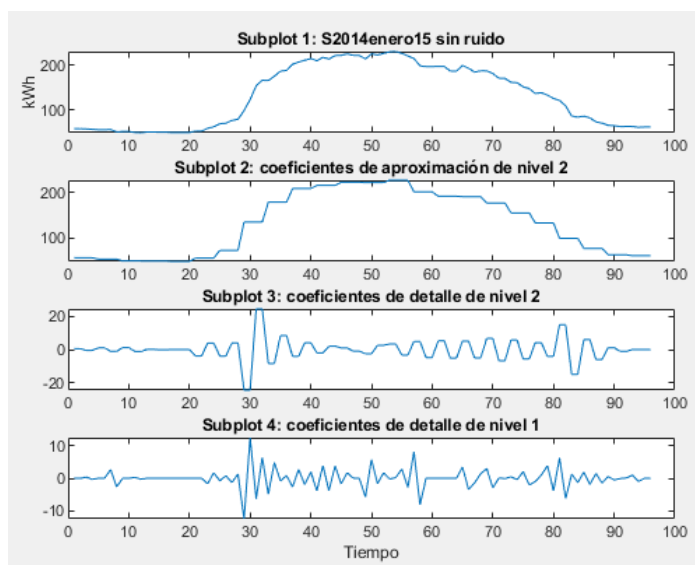


Figura 5.18. Descomposición *S2014enero15d*

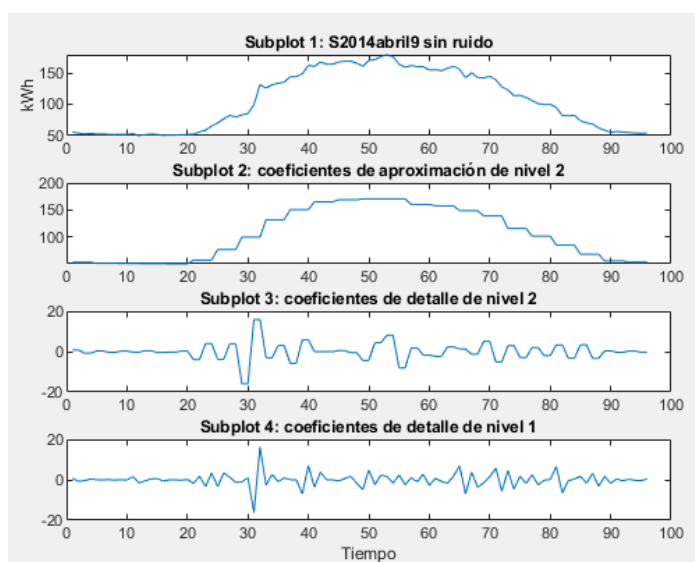


Figura 5.19. Descomposición *S2014abril9d*

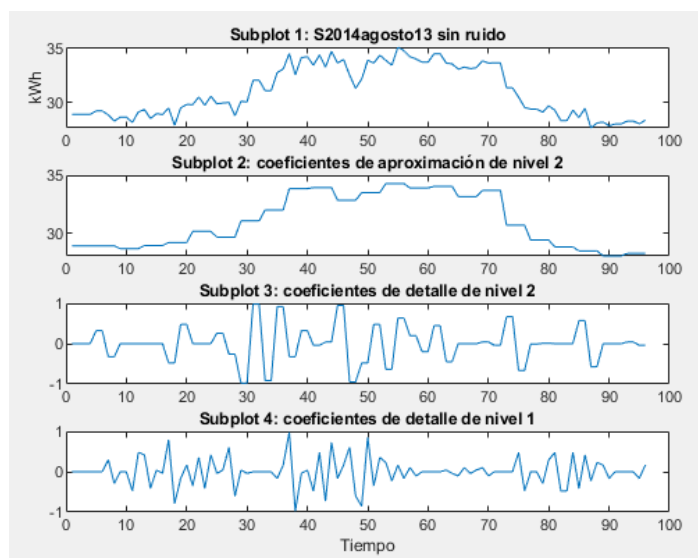


Figura 5.20. Descomposición S2014agosto13d

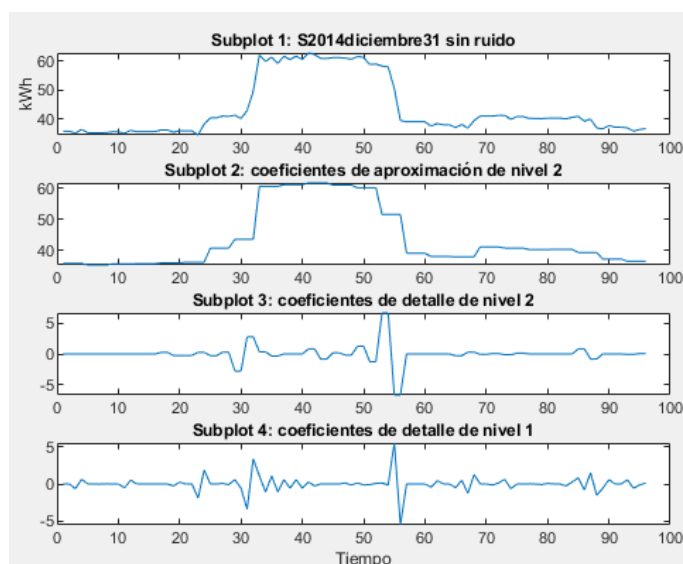


Figura 5.21. Descomposición S2014diciembre31d

#### 5.2.4. Transformada continua de Wavelet (CWT)

Procedemos a realizar la transformada continua de wavelet (CWT) para obtener los gráficos escala-tiempo de cada señal y de sus coeficientes de modo que podamos ver su comportamiento frecuencial a medida que pasa el tiempo.

Para ello, utilizaremos las líneas de código en la “command window” de Matlab que se encuentran en el anexo A.1 en su totalidad.

### 5.3. Datos obtenidos y conclusiones

Los coeficientes de las wavelets son el resultado de la integral de convolución entre la señal y la wavelet debidamente trasladada y ampliada en función de la escala en que estemos. A su vez, hay que tener en cuenta que estamos usando la wavelet de haar, que dará signos opuestos a las dos mitades del trozo que esté analizando. Por tanto, cuanto más regular sea la función (más parecida a una recta horizontal), menores serán los coeficientes resultantes, ya que surgirán de la diferencia de dos valores prácticamente iguales. Los máximos coeficientes, así pues, se darán cuando la señal tenga un comportamiento creciente/decreciente significativo. Por lo tanto, podemos afirmar que los coeficientes son, en cierta medida, proporcionales a la pendiente media en la escala afectada.

Los valles de coeficientes prácticamente nulos nos indican simetría entre ambos lados del valle en la señal con respecto al tiempo.

Los coeficientes de detalle (cD) resultan de la aplicación de un filtro pasa-alta frecuencia, y los de aproximación (cA) de un filtro pasa-baja. Los coeficientes de detalle de nivel 1 concentran los cambios que suceden más rápidos, por tanto, decimos que concentra el contenido de alta frecuencia. En estos gráficos se espera encontrar la actividad que no dure más de 15 minutos (que es el mínimo tiempo del que disponemos entre medidas). Encontraremos la alta frecuencia que nos permite estudiar la señal.

#### 5.3.1. CWT de las señales sin descomponer

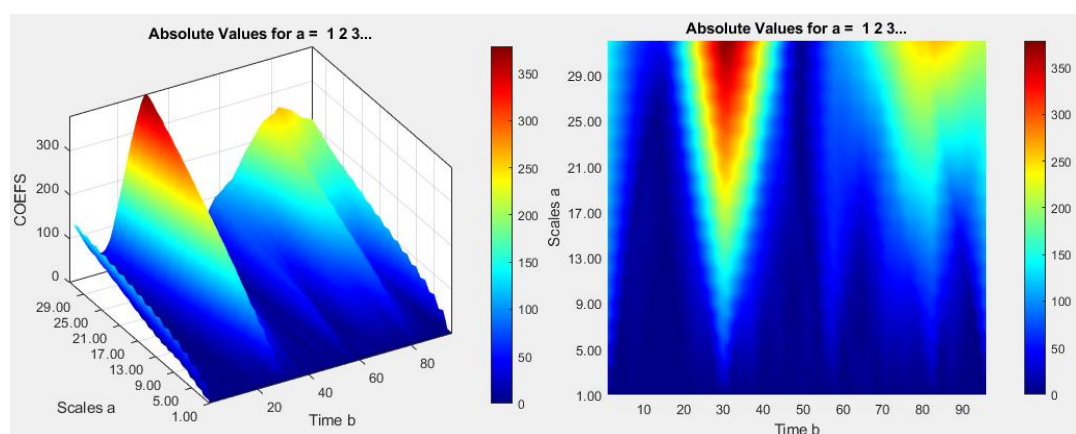


Figura 5.22. CWT(S2014enero15d) en 3D y en 2D

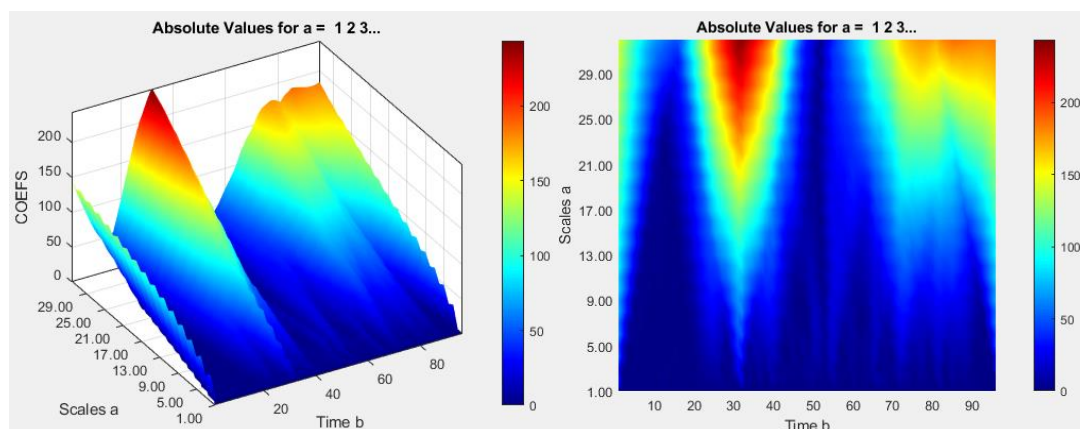


Figura 5.23. CWT(S2014abril9d) en 3D y en 2D

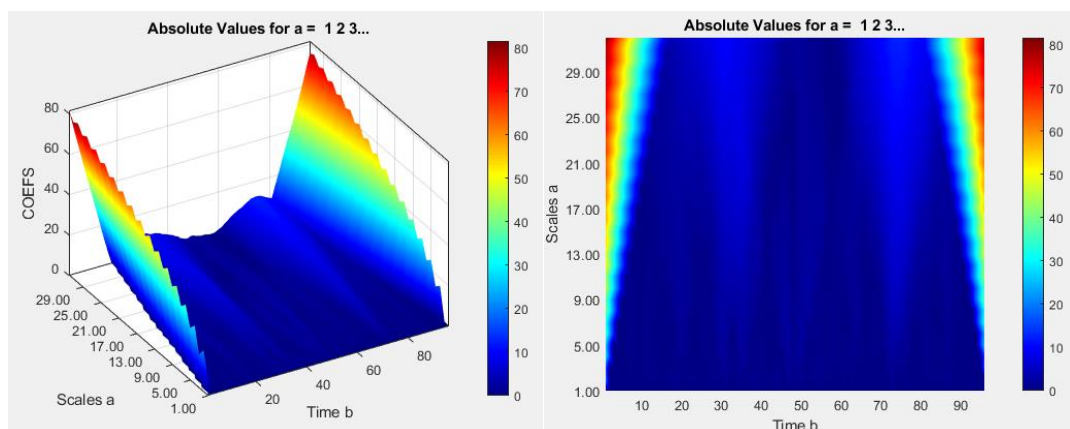


Figura 5.24. CWT(S2014agosto13d) en 3D y en 2D

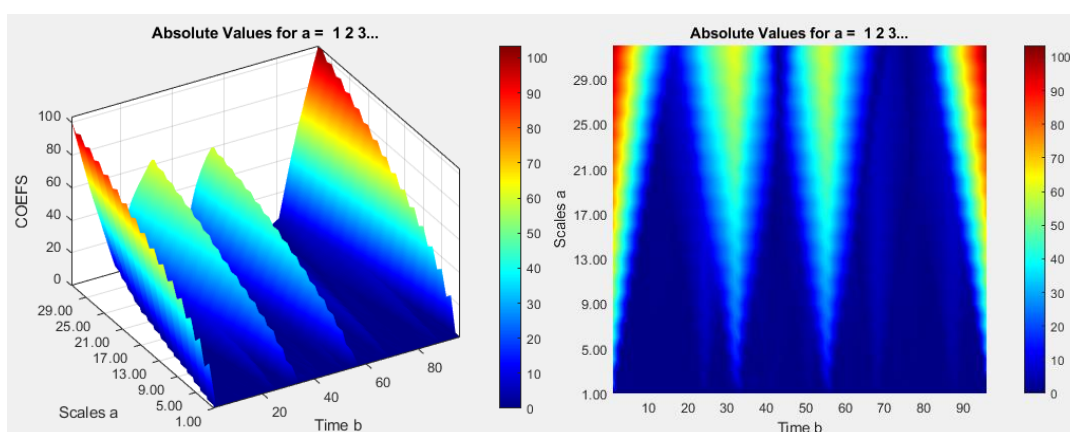


Figura 5.25. CWT(S2014diciembre31d) en 3D y en 2D

### 5.3.1.1. Observaciones (1)

- 15 de enero y 9 de abril

Los dos días lectivos tienen comportamientos parecidos, con un fuerte cono de influencia centrado en el valor 30 de la señal diaria, el cual corresponde a las 07:30h (recordamos que cada valor equivale a 15 minutos). Estos coeficientes altos corresponden a la subida drástica (véase figura 5.1) que experimenta el consumo de la universidad a esa hora. Después se nota la presencia de otros coeficientes elevados correspondientes al descenso del consumo que alcanza su máximo entre los valores 82 y 84 (20:30h-21:00h), ya que ahí están centrados y lo podemos ver fijándonos en las escalas inferiores, donde se hace un barrido de la señal más localizado en el tiempo. Puede observarse pues que la entrada en funcionamiento de aparatos es más brusca que el descenso del mismo, indicando un uso asimétrico a lo largo de la jornada.

Más allá de la diferencia de la magnitud de los coeficientes entre la señal de enero y la de abril, la cual está relacionada con el menor consumo de abril (hace que la diferencia existente entre dos puntos de un intervalo sea menor), para conocer más detalles sobre sus diferencias analizaremos sus coeficientes de detalle y de aproximación por separado que nos darán una información más detallada.

Por lo que respecta a los dos días festivos, sí que encontramos más diferencias en su comportamiento visibles en estos gráficos.

- 13 de agosto

Lo que llama la atención a primera vista son los dos extremos de la señal. Vemos inicio y final tintados de rojo exponiendo de esta forma que los máximos coeficientes se encuentran a principio y fin. Estos coeficientes corresponden a una representación de coeficientes ficticios de lo que realmente pasa en la señal, y es que, al igual que pasa con Fourier, se produce un efecto causado por el ventanado que distorsiona los coeficientes cercanos a los límites de la ventana (a más alta escala en más amplitud tiene efecto), y por esto el efecto abanico que se observa. Esto siempre sucede, pero cuando los coeficientes en el resto de la señal no son suficientemente grandes quedan ensombrecidos con respecto a este efecto.

La señal de agosto es bastante lineal, lo que nos hace esperar unos coeficientes bajos, y es por ello que destacan los extremos. Si nos fijamos en el centro del gráfico, podemos ver actividad en zonas a escalas altas y que se centran entre las 08:00h, 12:30h y 18:00h (valores 32, 50 y 72). Con la descomposición de la señal se analizará mejor lo que está sucediendo.

- 31 de diciembre



Aquí también cogen protagonismo los extremos, aunque no tanto, y nos permite ver qué sucede en el resto de la señal con más claridad. La principal diferencia con agosto es que la señal de diciembre tiene un consumo que crece a las 08:00h y se mantiene más o menos estable hasta las 14:00h (valores 32 y 56 respectivamente).

Esto lo podemos ver tanto en la imagen de la CWT como en la figura 5.1. Por ello para conocer más a fondo las señales analizaremos a continuación sus vectores de coeficientes surgidos de la descomposición realizada.

### 5.3.2. CWT de los coeficientes de detalle de nivel 1 (cd1r)

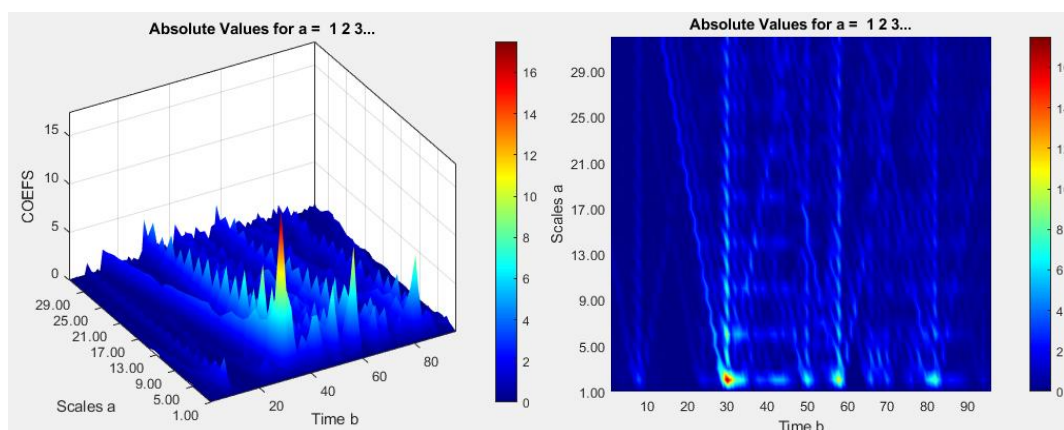


Figura 5.26. CWT(S2014enero15d-cd1r) en 3D y en 2D

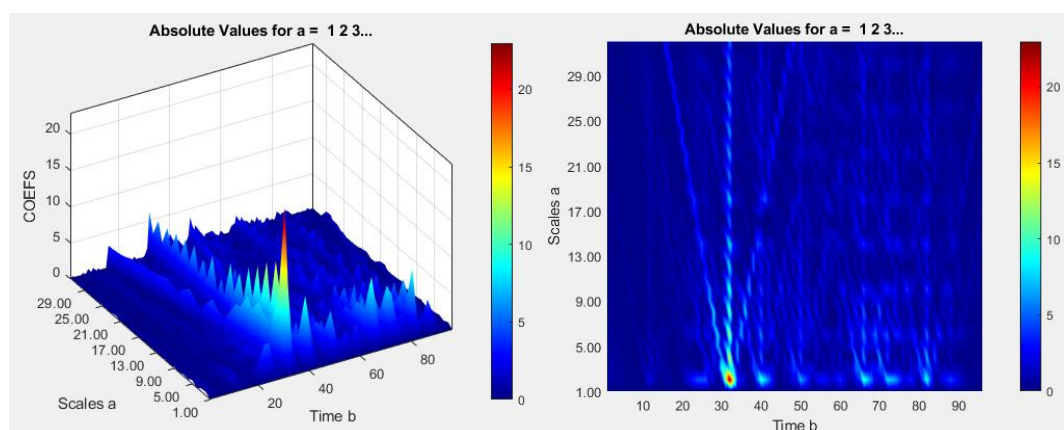


Figura 5.27. CWT(S2014abril9d-cd1r) en 3D y en 2D

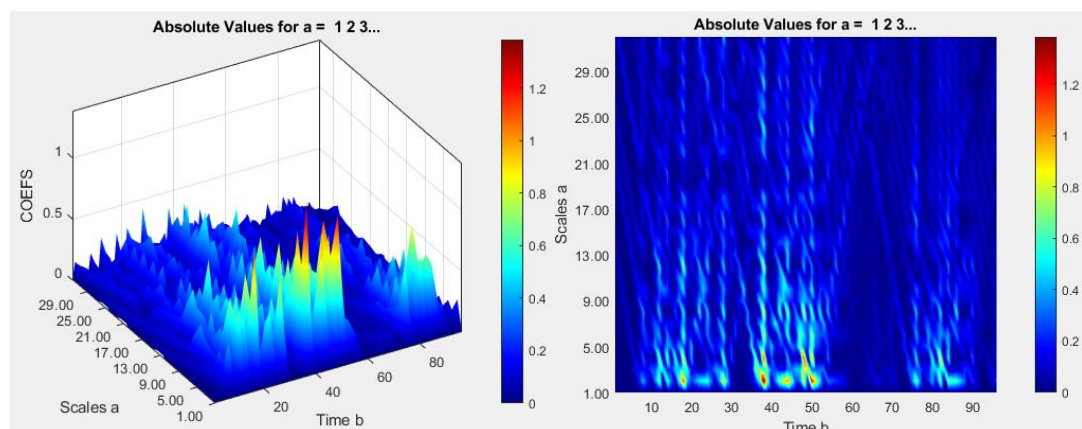


Figura 5.28. CWT(S2014agosto13d-cd1r) en 3D y en 2D

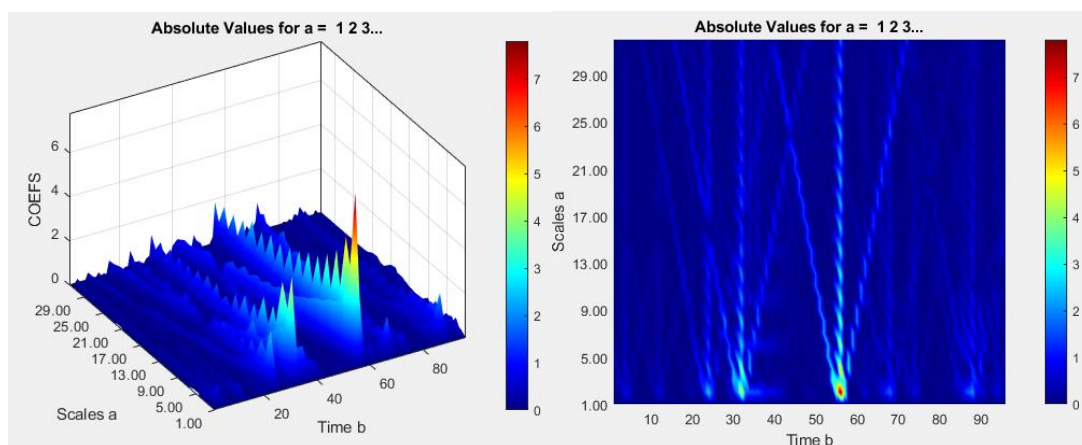


Figura 5.29. CWT(S2014diciembre31d-cd1r) en 3D y en 2D

### 5.3.2.1. Observaciones (2)

- 15 de enero

El máximo coeficiente se da a las 07:30h y se debe a la rápida subida del consumo. Si nos fijamos en el gráfico en tres dimensiones vemos como se dan tres picos pero que van decreciendo. El pico más alto es a las 07:30 de la mañana, el siguiente es a las 14:30h y el tercero a las 20:30h.

Es importante apuntar que fuera de horario de funcionamiento de la universidad (07:30-20:30h) no se aprecian coeficientes elevados, aunque si hay algo de presencia entre los valores señalados como máximos. Fuera de este horario la presencia de coeficientes elevados es mínima.

- 9 de abril

Es interesante señalar que el valor máximo de coeficientes de este día de abril es mayor al de enero, aun teniendo menor consumo. Se debe a que ha habido, a las 07:30h, un cambio en el consumo más pronunciado y brusco, con una pendiente más elevada.

Por lo general, los coeficientes se ajustan al comportamiento que ha presentado el 15 de enero, los coeficientes máximos corresponden a las mismas horas y la presencia de grandes coeficientes se concentran en horario lectivo.

- 13 de agosto

Este día sí que difiere de los demás. Vemos un comportamiento mucho más homogéneo y uniformemente distribuido. Los coeficientes son bastante más bajos que en días lectivos, y nos hace suponer que en esos días lectivos puede estar sucediendo este comportamiento, pero queda ensombrecido por los demás coeficientes de mayor valor.

- 31 de diciembre

Vemos los grandes coeficientes centrados en las horas 08:00h y 14:30h. Desaparece el coeficiente de alto valor de las 20:30h ya que no sigue el horario de los días lectivos, sino que ese día la universidad abrió solo por la mañana los servicios básicos (biblioteca, recepción...). Los coeficientes tienen valores de 0 a 7, lo cual está relacionado con la mayor pendiente que tiene la subida de consumo a las 8 de la mañana.

Para hacer un estudio donde veamos el comportamiento de alta frecuencia real no podemos tener una serie de datos con un paso de 15 minutos entre medidas. En este estudio estamos hablando de una pseudo-frecuencia o frecuencia relativa, que está condicionada por el hecho de que entre un valor y otro ha pasado un cuarto de hora. Para poder observar si realmente esta frecuencia responde a algún tipo de respuesta del consumo, o si bien es puramente debido al efecto del muestreo, deberíamos poder obtener datos con mayor frecuencia de muestreo.

Concluimos que las máximas pendientes se dan en horas de inicio y final del horario que tiene la universidad, con importancia concreta a las 14:00h del 15 de enero que hay una bajada de consumo con alta pendiente negativa.

Además, observando la señal de agosto, donde no hay prácticamente actividad, intuimos que existe una presencia de alta frecuencia más o menos distribuida en el tiempo, a excepción de la franja horaria de 13:00h a 19:30h, pero que el aporte es mínimo.

### 5.3.3. CWT de los coeficientes de detalle de nivel 2 (cd2r)

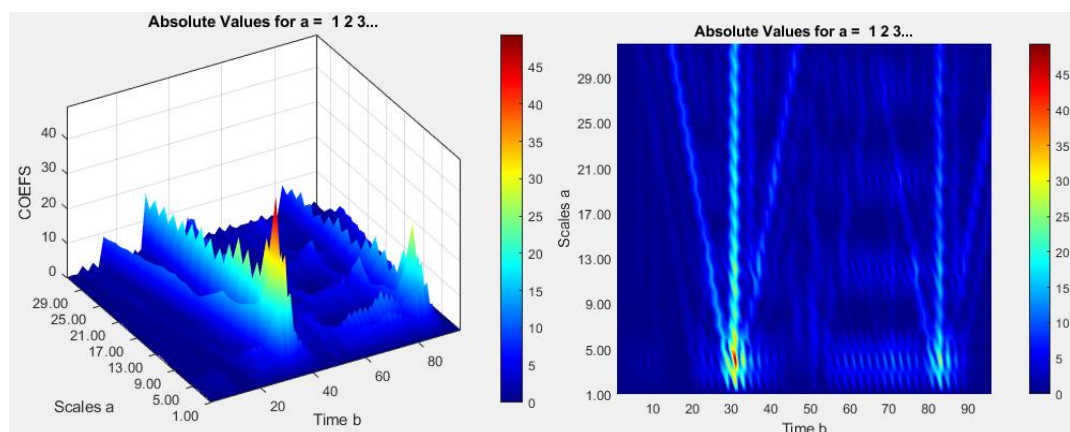


Figura 5.30. CWT(S2014enero15d-cd2r) en 3D y en 2D

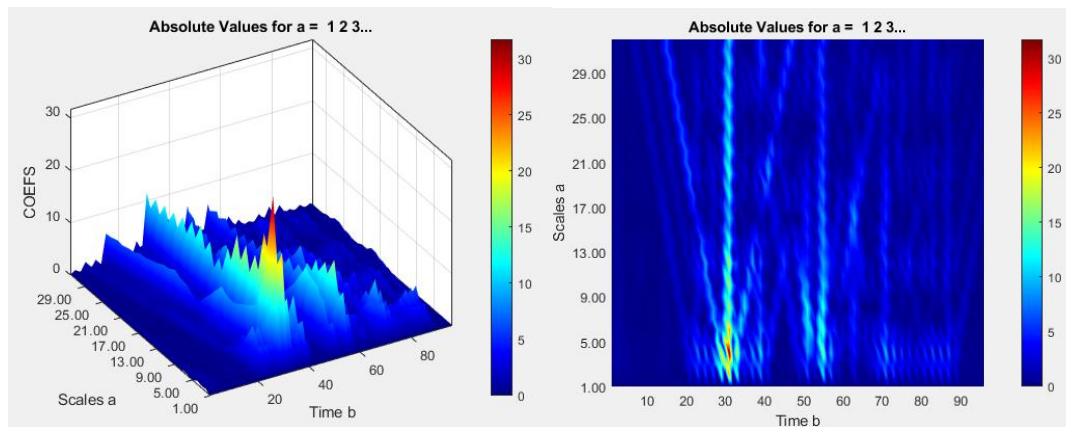


Figura 5.31. CWT(S2014abril9d-cd2r) en 3D y en 2D

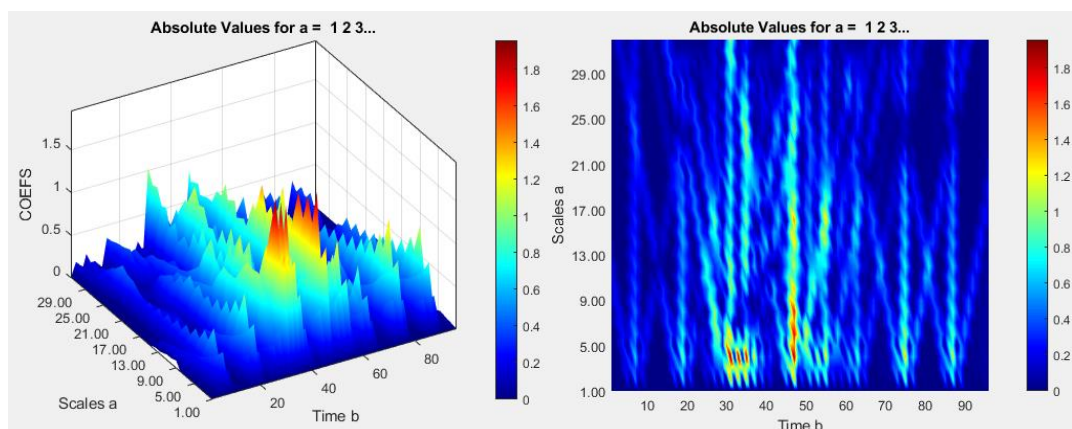


Figura 5.32. CWT(S2014agosto13d-cd2r) en 3D y en 2D

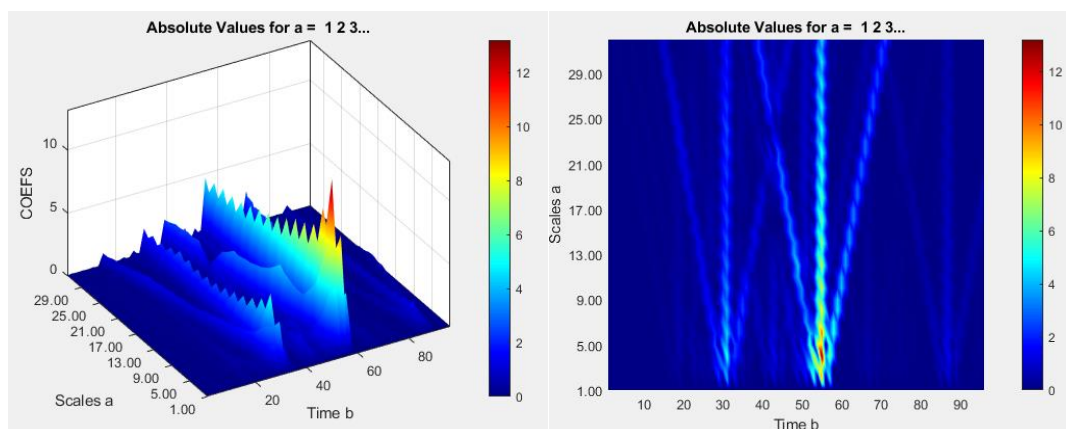


Figura 5.33. CWT(S2014diciembre31d-cd2r) en 3D y en 2D

### 5.3.3.1. Observaciones (3)

- 15 de enero 2014

Podemos encontrar dos diferencias significativas. Una es que el componente de las 14:30h que aparece en cd1r ha desaparecido en cd2r. Los picos de las 07:30h y 20:30h se mantienen, aunque son visibles en más escalas por la forma en que se descompone la señal en el nivel dos (una vez recompuesta, se ensancha temporalmente el vector de coeficientes y cada uno tiene una duración mayor).

Otra diferencia es que los coeficientes han subido de valor en el nivel 2. Esto se debe a que los coeficientes de más alta frecuencia son más en cantidad y por ello la energía se reparte entre todos ellos y resulta menor su importancia por coeficiente. Además, al estar más localizados en el tiempo, pierden importancia en la recomposición de la señal a nivel general.

Podemos concluir que la señal debe gran parte de su comportamiento a la baja frecuencia, y no es vinculante directamente al consumo eléctrico de la universidad, sino a que la medida de datos cada 15 minutos no nos permite analizar frecuencias más altas. Vemos actividad cada 30 minutos en la escala 4, que tiene imagen en escalas 12, 20 y 28 ( $4n$ ,  $n=1,3,5,7$ ).

- 9 de abril 2014

En este caso se mantiene el pico de las 07:30h, pero varía en el resto. Aparecen centrados a las 14:00h coeficientes grandes en cd2r que en cd1r no se han detectado. Vemos el periodo de 30 minutos en los coeficientes elevados y en las mismas escalas que sucedía en el mes de enero.

- 13 de agosto 2014

Coeficientes similares, un poco mayores en cd2r. Podemos observar que hay cierta compensación entre las dos señales que no se daba tan claramente en las señales de los días lectivos. Se ve como en intervalos de tiempo en los que predomina cd1r no hay tanta actividad en cd2r y viceversa.

- 31 de diciembre 2014

Los coeficientes más grandes que se dan en horas de apertura/cierre hacen que los coeficientes que pueden ocurrir en otros momentos queden ocultos, pero es un periodo de no actividad y ya sabemos qué pasa (ver señales de agosto). Destaca que, por primera vez, la señal de las 14:00h supere a la de las 08:00h. No varía en demasía cd2r respecto cd1r.



### 5.3.4. CWT de los coeficientes de aproximación de nivel 2 (a2r)

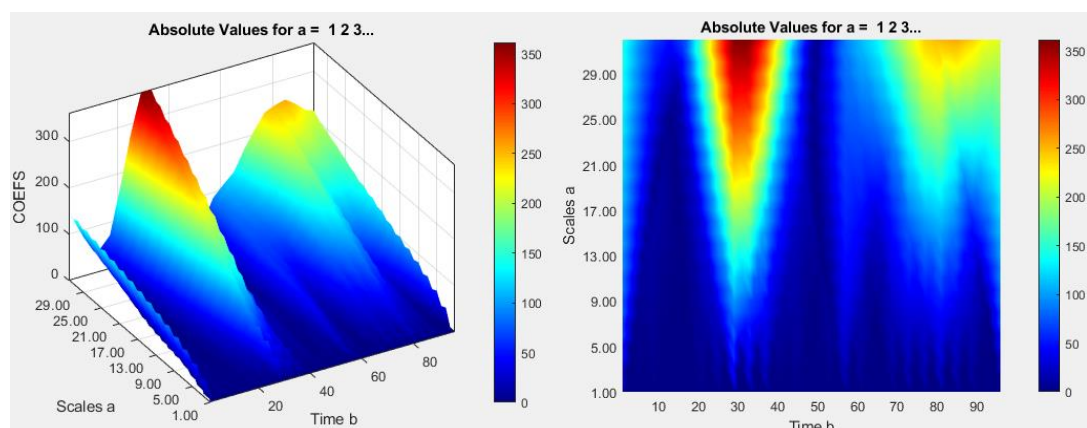


Figura 5.34. CWT(S2014enero15d-a2r) en 3D y en 2D

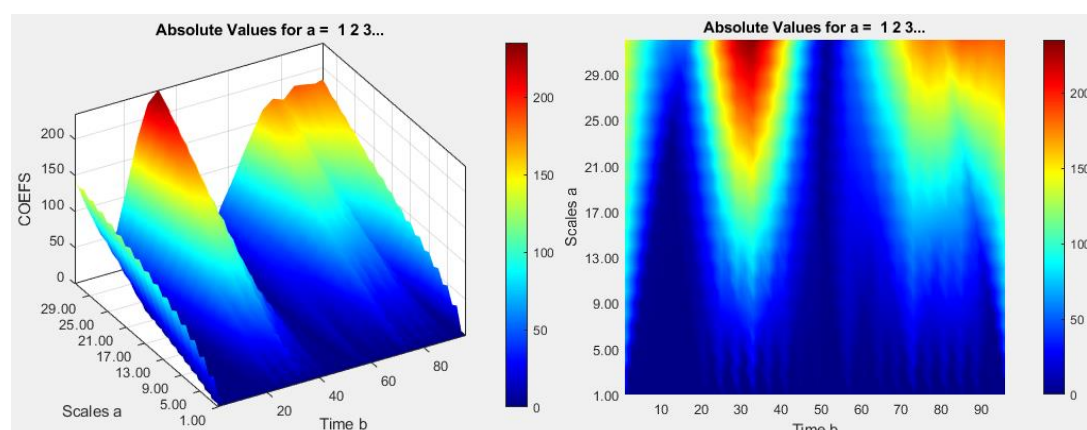


Figura 5.35. CWT(S2014abril9d-a2r) en 3D y en 2D

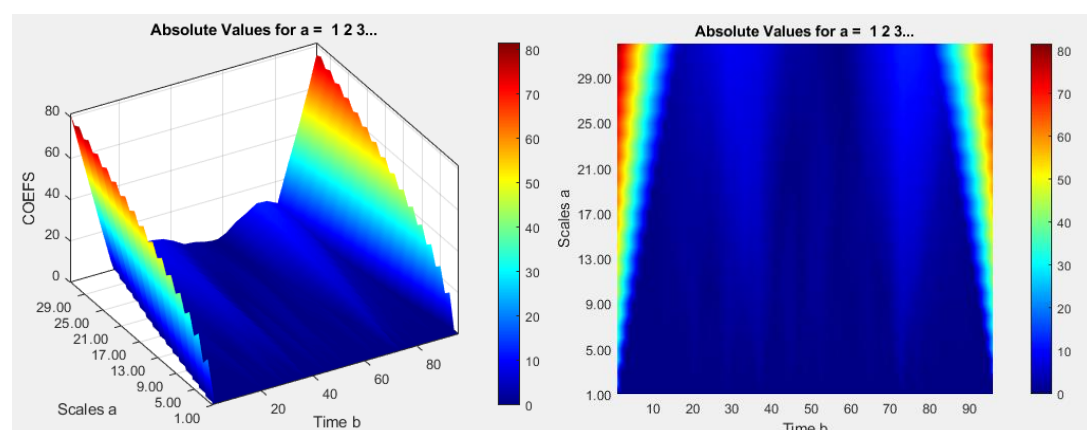


Figura 5.36. CWT(S2014agosto13d-a2r) en 3D y en 2D

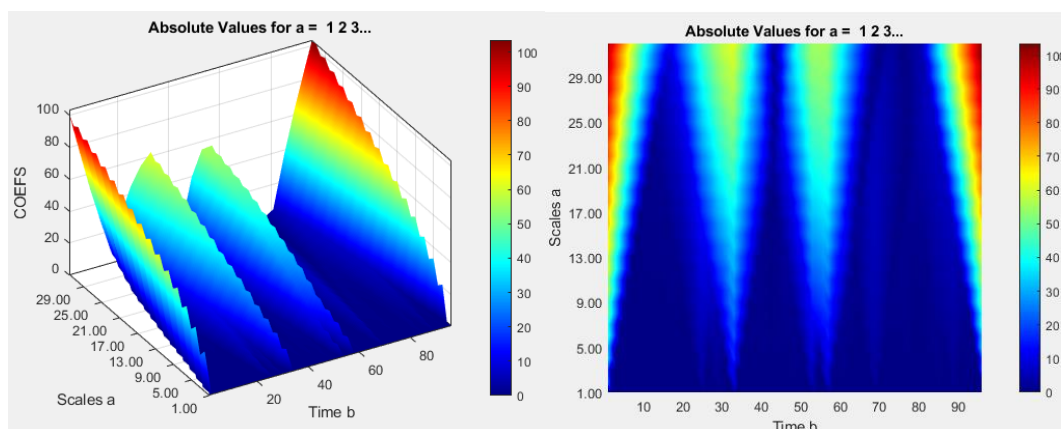


Figura 5.37. CWT(S2014diciembre31d-a2r) en 3D y en 2D

#### 5.3.4.1. Observaciones (4)

Los coeficientes de aproximación a2r son muy parecidos a los coeficientes resultantes de la señal sin descomponer, es decir que los coeficientes de aproximación se quedan la mayor parte de energía almacenada en la señal completa. Vemos diferencias en que aparece presencia de coeficientes en las escalas más bajas en la señal a2r, y eso es porque al quitar los coeficientes de los detalles (cd1r y cd2r) se produce una diferencia de valores que el gráfico, como está en valor absoluto, refleja.

Confirmamos así el comportamiento principalmente de baja frecuencia que contiene nuestra señal, condicionado por la frecuencia de muestreo que tiene.

#### 5.3.5. CWT de la suma de los coeficientes de detalle (cd1r+cd2r)

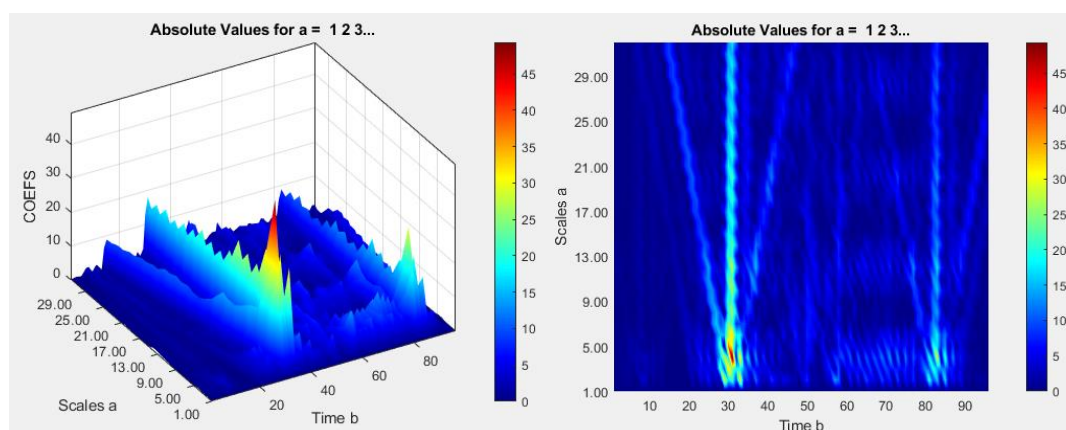


Figura 5.38. CWT(S2014enero15d-cd1r+cd2r) en 3D y en 2D



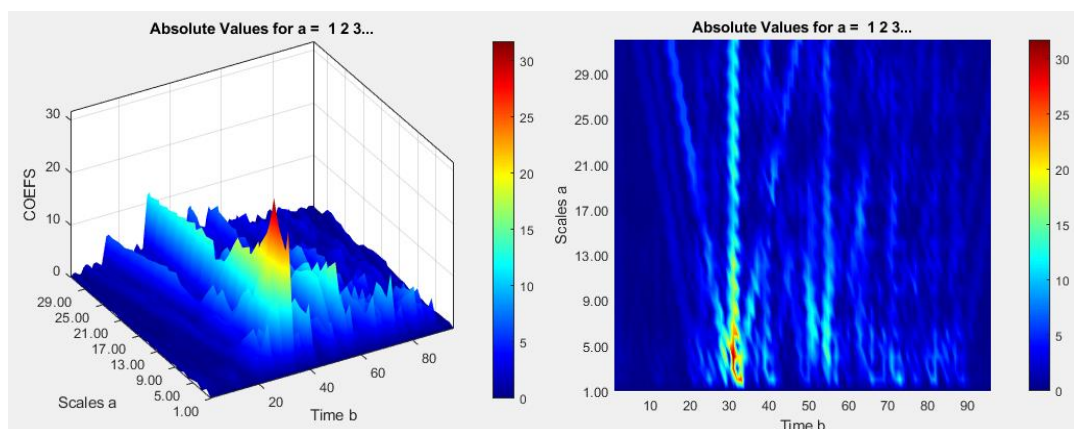


Figura 5.39. CWT(S2014abril9d-cd1r+cd2r) en 3D y en 2D

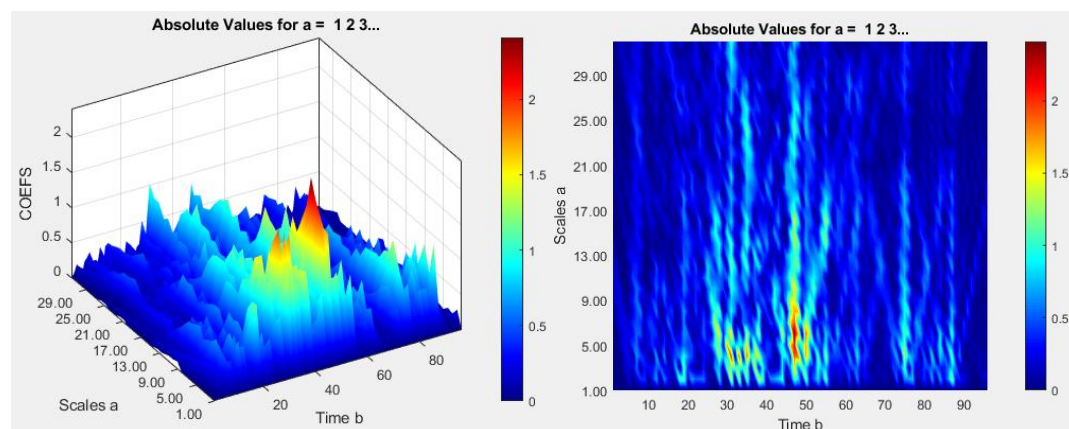


Figura 5.40. CWT(S2014agosto13d-cd1r+cd2r) en 3D y en 2D

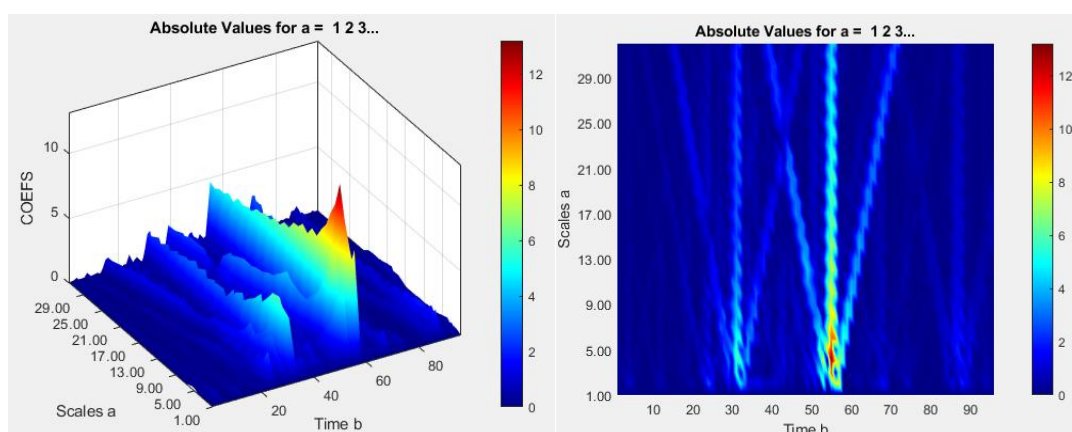


Figura 5.41. CWT(S2014diciembre31d-cd1r+cd2r) en 3D y en 2D

### 5.3.5.1. Observaciones (5)

El efecto que tienen los coeficientes de detalle con respecto a los de aproximación se puede observar con estos gráficos (cd1r+cd2r) o comparando los gráficos de a2r con la señal sin reconstruir (como ya hemos visto).

Observamos con más claridad que los detalles con valor más elevado se producen a las 07:30h, y es que al final del día el consumo decrece con menos pendiente. En los días festivos al medio día es cuando se observa un cambio más brusco del consumo.

Se tiene que relativizar los máximos coeficientes en cada día estudiado ya que, aun teniendo diferentes valores, son máximos en su contexto, y si se produce un cambio rápido que llegue a un consumo inferior entre un mes y otro, los coeficientes tendrán diferente valor, pero serán máximos igual y hay que tratarlos de manera similar.

### 5.3.6. Valoraciones finales

Después de analizar las transformadas realizadas, podemos concluir los siguientes puntos:

- La señal presenta un claro comportamiento de baja frecuencia, probablemente condicionado por la frecuencia de muestreo de los datos (1 valor cada 15 minutos).
- En periodo vacacional resultan unos coeficientes de detalle muy pequeños y, dentro de éstos, los máximos están repartidos a lo largo de la señal.
- El momento más crítico donde convive más contenido frecuencial distinto es las 07:30h. Es el horario de apertura de la universidad y se produce el aumento de consumo más importante y brusco del día.
- Cuando no hay actividad en la universidad, no se producen variaciones significativas en la señal.
- A grandes rasgos, no se detecta gran diferencia entre los valores en que se centran los coeficientes en el análisis a los coeficientes de detalle con los del análisis de los coeficientes de aproximación.
- El método utilizado para realizar el thresholding (Heuristic SURE) no es eficaz para trabajar con señales de consumo en días festivos, ya que modifica gran parte de la señal. Posiblemente los coeficientes que selecciona como residuales contengan poca energía de la señal, pero nos parece que la señal resultante no es fiel a la señal original y que se ha perdido mucha información.
- Los coeficientes de detalle de nivel 2 no nos dan información adicional valiosa con respecto a los de nivel 1, y es por ello por lo que en una señal de consumo diaria puede ser innecesario una descomposición de más de 1 nivel.
  - Es posible que cd1 no retenga la alta frecuencia esperada con respecto a cd2 por culpa de la baja frecuencia de muestreo de la señal.
- Los gráficos de la CWT nos muestran una información que es fácilmente relacionable mirando las señales iniciales consumo-tiempo, pero también información no tan fácil de identificar a primera vista.

- Para analizar tendencia de frecuencia menor sería interesante analizar periodos superiores a un día.

## 6. Modelización para la predicción de consumos

### 6.1. Explicación del modelo predictivo a utilizar

Uno de los objetivos vinculados al análisis del consumo consiste en realizar una predicción del consumo de un día a partir de los datos de consumo del día anterior. Es decir, predecir el consumo de mañana conociendo el consumo de hoy. Con una predicción del consumo, los servicios de mantenimiento podrán evaluar anomalías con mayor velocidad, y, si es pertinente, actuar para evitar funcionamientos incorrectos. Por otro lado, con una predicción a largo plazo sería más fácil analizar necesidades futuras y establecer políticas energéticas que permitan reducir el coste del consumo.

En el proyecto que precede a este, se intentó realizar un algoritmo de predicción basado en el análisis de tendencias en series de datos. El resultado del algoritmo daba buenos resultados para la mayoría de los días, aunque se alejaba mucho al aparecer comportamientos diferenciados.

El objetivo de este capítulo es el desarrollo de un algoritmo de predicción basado en wavelets que, posteriormente, pueda ser modificado para evaluar diferentes estrategias de predicción, y poder evaluar así si la capacidad de descomposición y detección de anomalías de las wavelets permite corregir los errores del algoritmo basado en tendencias. No es por tanto objeto de este proyecto depurar el algoritmo hasta encontrar una predicción altamente fiable, sino solo la implementación del mismo para su posterior depuración en la estrategia de selección de componentes.

Las wavelets tienen un potencial de descomposición que puede ser muy útil a la hora de encontrar coincidencias con otros días. En el algoritmo desarrollado, supondremos que las componentes frecuenciales se comportan independientemente. La idea será encontrar los días que se parezcan más en cuanto a su comportamiento de baja frecuencia, y utilizar esos datos para nuestra predicción de baja frecuencia. Para el comportamiento de alta frecuencia se realizará el mismo proceso. Al final tendremos coeficientes de muchos días diferentes y cuando recompongamos nos quedará una señal que será mezcla de todos ellos.

Se encontrarán los valores de los cuatro coeficientes más cercanos a los del día de 'hoy' según su tipo (cd1, cd2 o a2). Una vez los sepamos y los situemos en el tiempo, se cogerán las señales de consumo del día siguiente de cada uno de ellos y se hará un promedio. Después se recompondrá la señal con estos promedios y esa sería nuestra predicción.

Para encontrar la cercanía entre los vectores se aplicará un sumatorio de las diferencias cuadráticas de cada coeficiente.

## 6.2. Procedimiento de predicción

Haremos la prueba con el miércoles 14 de enero de 2015 y predeciremos el jueves 15 de enero de 2015.

Tomaremos como histórico el año 2014, y los pasos a seguir son:

1r: Definir la variable S2015enero14 y descomponerla en 2 niveles.

2n: Reconponer los coeficientes por separado.

3r: Crear 4 variables de los 4 primeros miércoles del año 2014 y descomponerlos. Encontrar la distancia que separa cada miércoles de 2014 con el miércoles de 2015 y tendremos 4 valores de distancia por cada coeficiente (es decir, si hacemos 2 niveles, tendremos cd1, cd2 y a2, por tanto 12 valores de distancia (3 valores por día)).

4t: Crear el 5t miércoles de 2014 y calcular la distancia (de cada nivel por separado) con el miércoles de 2015. Si la distancia con algún coeficiente es mayor que las primeras 4, descartar este día. Si es menor que alguna de ellas, sustituirla. Al final nos quedaremos con las 4 distancias más pequeñas de cada nivel. (12) (tendremos los 4 miércoles de 2014 más cercanos al nuestro de 2015).

5t: Repetir el punto 4 hasta completar las 52 semanas del año 2014.

6t: Sabiendo qué días son los más cercanos, mirar el día siguiente (jueves) de cada uno de estos días. Descomponer los 4 jueves (de cada nivel) de 2014 y hacer un promedio de los coeficientes que nos de cada día. (es posible que cuando hablemos de cd1 nos dé días de meses diferentes que cuando hablemos de cd2 o de a2).

7t: Tendremos finalmente 3 vectores de coeficientes cd1, cd2 y a2. Reconponemos y será el consumo de este jueves de 2015.

8t: Comprobar la predicción con el real.

### 6.3. Datos obtenidos y conclusiones

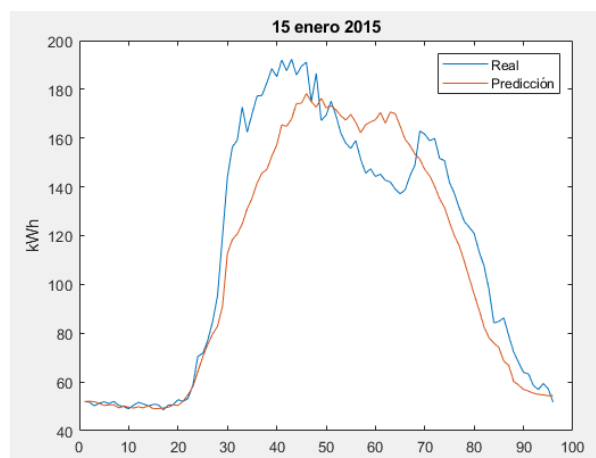


Figura 6.1. Consumo obtenido y consumo real (Fuente: Elaboración propia)

semanascd1				semanascd2				semanasa2			
41	44	42	45	6	21	15	47	45	47	12	48

Figura 6.2. Semanas de los días más parecidos según tipo de coeficientes

rmsreal	rmsprediccion
125.95	118.97

Figura 6.3. Valor eficaz de la señal real y de la predicción

La señal no se adapta como nos hubiera gustado a la realidad, aunque sí lo hace el valor eficaz de las dos señales que se encuentran en un punto similar.

Es posible que una descomposición a más niveles se ajuste más a la realidad, o quizás las técnicas para encontrar la cercanía (sumatorio de diferencias cuadráticas) o el promediado utilizado al final no sean tan efectivos.

Aun así, observamos cómo la señal obtenida sí que sigue una tendencia similar a la real, y lo que predice como consumo mayor de lo que realmente sucede lo acaba compensando en otro momento que predice un consumo menor.

El código empleado se encuentra en el anexo A.2.

## 7. Propuestas de continuidad del trabajo

Para seguir, mejorar y complementar el estudio de los datos de consumo eléctrico de edificios universitarios o de otra naturaleza se recomienda probar las siguientes propuestas:

- Estudiar las señales con wavelets distintas a Haar para ver si se adaptan mejor y concentran los componentes de las señales en menos coeficientes.
- Aplicación de otro método de de-noising, que puede ser uno de los mencionados en el presente trabajo u otro método manual, para separar mejor señal y ruido.
- Trabajar con señales de mayor longitud temporal para permitir la detección de patrones interdiarios.
- Trabajar con señales con una frecuencia de muestreo más elevada para, entre otras cosas, observar si los límites de pseudo-frecuencia observadas son realmente debidos a la señal o a efectos de la frecuencia de muestreo.

Para mejorar el procedimiento de predicción se propone lo siguiente:

- Utilizar otra wavelet para hacer la descomposición en coeficientes
- Utilizar distintos métodos para medir la cercanía o distancia existente entre vectores de coeficientes.
- Optimización del código empleado y utilización de más años como histórico.
- Analizar si el uso de componentes frecuenciales de días distintos es el método adecuado para el análisis y, en caso contrario, evaluar si el análisis wavelet permite aportar algún tipo de mejora a la predicción basada en tendencias.

## Conclusiones

A lo largo de este trabajo se ha estudiado la teoría que envuelve a las wavelets, el concepto de su funcionamiento y el potencial que puede tener. Esta labor, que queda recogida en la primera parte del trabajo, ha hecho posible tener claro el procedimiento que se quería seguir para estudiar el comportamiento frecuencial de las señales utilizadas, y finalmente llegar a la parte final donde se utiliza la codificación de la transformada de wavelet para el desarrollo de un algoritmo de predicción de señales, que puede ser aplicable a cualquier señal. Se destaca el esfuerzo extra que comporta para un estudiante de grado el abordaje de una teoría matemáticamente compleja como las wavelets.

Las deducciones y conclusiones a las que se han llegado a lo largo del desarrollo de este documento han sido expuestas en cada capítulo correspondiente, aunque por destacar una, decir que todo el trabajo ha sido condicionado por la baja frecuencia de muestreo que tiene la señal, y que sería interesante realizar esta metodología con señales que tengan datos que hayan sido recogidos con menor espacio de tiempo entre ellos, más aún hablando de comportamiento eléctrico. Se escogió trabajar con la wavelet de Haar por su característica de ser discontinua, pensando así que captaría mejor los cambios bruscos y comportamientos anómalos de la señal que aparecerían, pero como estos cambios se producen muy rápido, se pierde esta ventaja por la baja frecuencia de muestreo de la señal. Este límite de frecuencia viene marcado por la tasa de muestreo común de los contadores, relacionada con la frecuencia de tasación de las compañías. Como resultado del trabajo deducimos que los datos de contadores pueden no ser suficientes para este tipo de predicción y análisis, necesitando obtener los datos de analizadores de redes y no de contadores.

El proyecto es como una introducción al análisis de datos y una propuesta de procedimiento en concreto utilizando las wavelets. No intenta abordar desde su concepción el problema de una forma finalista, sino ser parte de un proceso para ser posteriormente mejorado o complementado.

El algoritmo de predicción abre una ventana de posibilidades para futuras aplicaciones, ya que, en lugar de utilizar métodos de regresión para predecir datos, utiliza la codificación de las señales según su contenido frecuencial y las compara, por separado, con un histórico de datos para luego recomponer una señal que puede contener información de muchas otras diferentes. Aunque los resultados obtenidos en el trabajo son mejorables, su utilización en otro tipo de consumos, como el consumo de agua o en conjuntos más grandes (consumo de ciudades) presenta resultados interesantes. El algoritmo desarrollado se muestra como un punto de partida para ajustar posteriormente su predicción o analizar las diferencias con este tipo de consumos.

Sería favorable continuar por este camino y realizar el estudio con algunas de las recomendaciones expuestas en el capítulo 7 de este trabajo.



## Presupuesto

En esta sección del trabajo se indica un análisis económico correspondiente a los costes de elaboración del documento, costes genéricos de ingeniería y material utilizado para tal fin.

Se puede adquirir una licencia de Matlab anual por un coste de 800€/año. Estimando en 1800 horas de trabajo anual, al proyecto le corresponderían 249€.

El coste del ordenador portátil es de 700€, suponiéndole una vida útil de 5 años, y 1800h de trabajo anual, el precio vinculado al proyecto es de 43,5€. Además, el uso de Microsoft Office (7€/mes) durante 8 meses, suma 56€ adicionales.

El coste de transporte se asocia a los trayectos realizados para acudir a las reuniones de seguimiento y para acudir a la biblioteca a trabajar. Se ha acudido casi a diario y es por ello que se han utilizado tres bonos trimestrales (t-jove de TMB) con un coste de 105€ cada uno.

### **PRESUPUESTO**

	Dedicación [h]	Coste [€/h]	Coste [€]
<i>Búsqueda de información</i>	200	20	4000
<i>Elaboración del estudio</i>	200	20	4000
<i>Desarrollo del algoritmo predictivo</i>	40	20	800
<i>Elaboración de la memoria</i>	100	20	2000
<i>Reuniones</i>	20	20	400
<i>Dedicación total</i>	560	20	11200
<i>Licencia Matlab R2018b</i>	-	-	249
<i>Ordenador portátil ASUS X555UJ</i>	-	-	43,5
<i>Costes de transporte</i>	-	-	315
<i>Microsoft Office 365</i>	-	-	56
<i>IVA</i>			21%
<i>Coste total (sin impuestos)</i>			11863,5
<i>Coste total (con impuestos)</i>			14354,84

*Este presupuesto tiene una validez de 6 meses a partir del día 28-04-2019.*

## Bibliografía

1. Garzón Àngel M. Disseny d'un model per l'estimació del consum elèctric i la seva implementació en R. 2018.
2. Bonatti PA, Sauro L. A semantic stateless service description language. *CEUR Workshop Proc.* 2009;434.
3. Time Frequency Analysis & Wavelets. <https://www.youtube.com/watch?v=ViZYXxuxUKA>. Accessed April 22, 2019.
4. Olkkonen J. *DISCRETE WAVELET TRANSFORMS - THEORY AND APPLICATIONS* Edited by Juuso Olkkonen.; 2011.
5. "Continuous Wavelet Transform.". Wikipedia: La enciclopedia libre. s.f. [Fecha de consulta: 23/02/19]. [https://en.wikipedia.org/wiki/Continuous\\_wavelet\\_transform](https://en.wikipedia.org/wiki/Continuous_wavelet_transform). Published 2019.
6. Vázquez L, Cerezo ER. *Wavelets Proyecto Sistemas Informáticos*. [http://eprints.ucm.es/16705/1/SI\\_Elena\\_Rupérez\\_Cerezo\\_Jun12.pdf](http://eprints.ucm.es/16705/1/SI_Elena_Rupérez_Cerezo_Jun12.pdf). Accessed April 22, 2019.
7. Wavelet Toolbox - MATLAB. <https://es.mathworks.com/products/wavelet.html>. Accessed April 22, 2019.
8. "Discrete Wavelet Transform.". Wikipedia: La enciclopedia libre. s.f. [Fecha de consulta: 15/02/19]. [https://en.wikipedia.org/wiki/Discrete\\_wavelet\\_transform](https://en.wikipedia.org/wiki/Discrete_wavelet_transform).
9. "Haar Wavelet.". Wikipedia: La enciclopedia libre. s.f. [Fecha de consulta: 15/09/18]. [https://en.wikipedia.org/wiki/Haar\\_wavelet](https://en.wikipedia.org/wiki/Haar_wavelet).
10. "Wavelet.". Wikipedia: La enciclopedia libre. s.f. [Fecha de consulta: 25/09/18]. <https://en.wikipedia.org/wiki/Wavelet>.
11. Calendarios laborales de España por provincias. <https://www.calendarioslaborales.com/>. Accessed April 22, 2019.
12. Calendarios Académicos — Servei de Gestió Acadèmica — UPC. Universitat Politècnica de Catalunya. <https://www.upc.edu/sga/es/normativas/CalendariosAcademicos>. Accessed April 22, 2019.
13. Santa María F, Kouro Rodrigo Musalem M E SR. *Técnicas Modernas En Automática Tutorial Introductorio a La Teoría de Wavelet*. <http://www2.elo.utfsm.cl/~elo377/documentos/Wavelet.pdf>. Accessed April 22, 2019.
14. *CAPÍTULO 3 ALGORITMOS PARA REDUCCIÓN DE RUIDO EN SEÑALES*. [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lem/hernandez\\_d\\_m/capitulo3.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/hernandez_d_m/capitulo3.pdf). Accessed April 22, 2019.
15. Cohen R. *Signal Denoising Using Wavelets*.; 2012. <http://tx.technion.ac.il/~rc>. Accessed April 22, 2019.

16. Nason GP. *Choice of the Threshold Parameter in Wavelet Function Estimation*. <https://people.maths.bris.ac.uk/~magpn/Research/unpubpapers/gren.pdf>. Accessed April 22, 2019.
17. Gonz JW. WAVELET PARA EL ANÁLISIS DE TRANSITORIOS DEBIDOS A LA CONMUTACIÓN DE BANCOS DE .... 2010;(JANUARY).



## Anexo A

### A1. Comandos Matlab

Es importante que se realice el código por bloques de días y de coeficientes, ya que se utilizan las mismas variables y si se quiere compilar todo a la vez las variables se quedarán con el valor final que hayan adquirido.

#### - Capítulo 5.2.3.

```
%15 de enero 2014
>> [c,l]=wavedec(S2014enero15d,2,'haar');
[cd1,cd2]=detcoef(c,l,[1 2]);
a2=appcoef(c,l,'haar',2);
cd1r=wrcoef('d',c,l,'haar',1);
cd2r=wrcoef('d',c,l,'haar',2);
a2r=wrcoef('a',c,l,'haar',2);
subplot(4,1,1);
plot(S2014enero15d);
title('Subplot 1: S2014enero15 sin ruido');
ylabel('kWh');
subplot(4,1,2);
plot(a2r);
title('Subplot 2: coeficientes de aproximación de nivel 2');
subplot(4,1,3);
plot(cd2r);
title('Subplot 3: coeficientes de detalle de nivel 2');
subplot(4,1,4);
plot(cd1r);
title('Subplot 4: coeficientes de detalle de nivel 1');
xlabel('Tiempo');
```

```
%9 de abril 2014
>> [c,l]=wavedec(S2014abril9d,2,'haar');
[cd1,cd2]=detcoef(c,l,[1 2]);
a2=appcoef(c,l,'haar',2);
cd1r=wrcoef('d',c,l,'haar',1);
cd2r=wrcoef('d',c,l,'haar',2);
a2r=wrcoef('a',c,l,'haar',2);
subplot(4,1,1);
plot(S2014abril9d);
title('Subplot 1: S2014abril9 sin ruido');
ylabel('kWh');
subplot(4,1,2);
plot(a2r);
title('Subplot 2: coeficientes de aproximación de nivel 2');
subplot(4,1,3);
plot(cd2r);
title('Subplot 3: coeficientes de detalle de nivel 2');
```

```

subplot(4,1,4);
plot(cd1r);
title('Subplot 4: coeficientes de detalle de nivel 1')
xlabel('Tiempo');

%13 de agosto 2014
>> [c,l]=wavedec(S2014agosto13d,2,'haar');
[cd1,cd2]=detcoef(c,l,[1 2]);
a2=appcoef(c,l,'haar',2);
cd1r=wrcoef('d',c,l,'haar',1);
cd2r=wrcoef('d',c,l,'haar',2);
a2r=wrcoef('a',c,l,'haar',2);
subplot(4,1,1);
plot(S2014agosto13d);
title('Subplot 1: S2014agosto13 sin ruido')
ylabel('kWh');
subplot(4,1,2);
plot(a2r);
title('Subplot 2: coeficientes de aproximación de nivel 2')
subplot(4,1,3);
plot(cd2r);
title('Subplot 3: coeficientes de detalle de nivel 2')
subplot(4,1,4);
plot(cd1r);
title('Subplot 4: coeficientes de detalle de nivel 1')
xlabel('Tiempo');

```

```

%31 de diciembre 2014
>> [c,l]=wavedec(S2014diciembre31d,2,'haar');
>> [cd1,cd2]=detcoef(c,l,[1 2]);
>> a2=appcoef(c,l,'haar',2);
>> cd1r=wrcoef('d',c,l,'haar',1);
>> cd2r=wrcoef('d',c,l,'haar',2);
>> a2r=wrcoef('a',c,l,'haar',2);
>> subplot(4,1,1);
>> plot(S2014diciembre31d);
>> title('Subplot 1: S2014diciembre31 sin ruido')
ylabel('kWh');
>> subplot(4,1,2);
>> plot(a2r);
>> title('Subplot 2: coeficientes de aproximación de nivel 2')
>> subplot(4,1,3);
>> plot(cd2r);
>> title('Subplot 3: coeficientes de detalle de nivel 2')
>> subplot(4,1,4);
>> plot(cd1r);
>> title('Subplot 4: coeficientes de detalle de nivel 1')
xlabel('Tiempo');

```

- Capítulo 5.2.4.

```
%15 de enero 2014
>> [c,l]=wavedec(S2014enero15d,2,'haar');
[cd1,cd2]=detcoef(c,l,[1 2]);
a2=appcoef(c,l,'haar');
%Procedemos a recomponer cada vector de coeficientes (cd1, cd2 y a2) ya
que no todos tienen la misma longitud debido al procedimiento de
descomposición de la transformada.
cd1r=wrcoef('d',c,l,'haar',1);
cd2r=wrcoef('d',c,l,'haar',2);
a2r=wrcoef('a',c,l,'haar',2);
%Aplicamos la transformada continua de wavelet
%La escala máxima nos la dicta la app WaveletAnalyzer en su pestaña
'Continuous wavelet 1-D'
%Para cd1r
cwt(cd1r,1:32,'haar','3Dplot');colormap jet
%Para cd2r
cwt(cd2r,1:32,'haar','3Dplot');colormap jet
%Para a2r
cwt(a2r,1:32,'haar','3Dplot');colormap jet
%Para cd1r+cd2r
cwt(cd1r+cd2r,1:32,'haar','3Dplot');colormap jet

%9 de abril 2014
>> [c,l]=wavedec(S2014abril9d,2,'haar');
[cd1,cd2]=detcoef(c,l,[1 2]);
a2=appcoef(c,l,'haar');
%Procedemos a recomponer cada vector de coeficientes (cd1, cd2 y a2) ya
que no todos tienen la misma longitud debido al procedimiento de
descomposición de la transformada.
cd1r=wrcoef('d',c,l,'haar',1);
cd2r=wrcoef('d',c,l,'haar',2);
a2r=wrcoef('a',c,l,'haar',2);
%Aplicamos la transformada continua de wavelet
%La escala máxima nos la dicta la app WaveletAnalyzer en su pestaña
'Continuous wavelet 1-D'
%Para cd1r
cwt(cd1r,1:32,'haar','3Dplot');colormap jet
%Para cd2r
cwt(cd2r,1:32,'haar','3Dplot');colormap jet
%Para a2r
cwt(a2r,1:32,'haar','3Dplot');colormap jet
%Para cd1r+cd2r
cwt(cd1r+cd2r,1:32,'haar','3Dplot');colormap jet

%13 de agosto 2014
>> [c,l]=wavedec(S2014agosto13d,2,'haar');
[cd1,cd2]=detcoef(c,l,[1 2]);
a2=appcoef(c,l,'haar');
%Procedemos a recomponer cada vector de coeficientes (cd1, cd2 y a2) ya
que no todos tienen la misma longitud debido al procedimiento de
descomposición de la transformada.
```

```

cd1r=wrcoef('d',c,1,'haar',1);
cd2r=wrcoef('d',c,1,'haar',2);
a2r=wrcoef('a',c,1,'haar',2);
%Aplicamos la transformada continua de wavelet
%La escala máxima nos la dicta la app WaveletAnalyzer en su pestaña
'Continuous wavelet 1-D'
%Para cd1r
cwt(cd1r,1:32,'haar','3Dplot');colormap jet
%Para cd2r
cwt(cd2r,1:32,'haar','3Dplot');colormap jet
%Para a2r
cwt(a2r,1:32,'haar','3Dplot');colormap jet
%Para cd1r+cd2r
cwt(cd1r+cd2r,1:32,'haar','3Dplot');colormap jet

%31 de diciembre 2014
>> [c,1]=wavedec(S2014diciembre31d,2,'haar');
[cd1,cd2]=detcoef(c,1,[1 2]);
a2=appcoef(c,1,'haar');
%Procedemos a recomponer cada vector de coeficientes (cd1, cd2 y a2) ya
que no todos tienen la misma longitud debido al procedimiento de
descomposición de la transformada.
cd1r=wrcoef('d',c,1,'haar',1);
cd2r=wrcoef('d',c,1,'haar',2);
a2r=wrcoef('a',c,1,'haar',2);
%Aplicamos la transformada continua de wavelet
%La escala máxima nos la dicta la app WaveletAnalyzer en su pestaña
'Continuous wavelet 1-D'
%Para cd1r
cwt(cd1r,1:32,'haar','3Dplot');colormap jet
%Para cd2r
cwt(cd2r,1:32,'haar','3Dplot');colormap jet
%Para a2r
cwt(a2r,1:32,'haar','3Dplot');colormap jet
%Para cd1r+cd2r
cwt(cd1r+cd2r,1:32,'haar','3Dplot');colormap jet

```



## A2. Algoritmo de predicción

```
>> load('workspaceTFG1.mat')
%creamos la señal de la variable de 'hoy' (miércoles 14 de enero de 2015)
>> S2015enero14=S2015(1249:1344);
%Creamos la señal real del día que queremos predecir ('mañana') para
hacer la comparación
S2015enero15=S2015(1345:1440);
%Descomposición
[c,1]=wavedec(S2015enero14,2,'haar');
[cd1,cd2]=detcoef(c,1,[1 2]);
a2=appcoef(c,1,'haar');
%recomponemos por separado
cd1r=wrcoef('d',c,1,'haar',1);
cd2r=wrcoef('d',c,1,'haar',2);
a2r=wrcoef('a',c,1,'haar',2);
%Creamos los 4 primeros miércoles del año 2014
S2014enero1=S2014(1:96);
S2014enero8=S2014(673:768);
S2014enero15=S2014(1345:1440);
S2014enero22=S2014(2017:2112);
%descomposición de los 4 días y recomposición de los coeficientes
resultantes
%Primer día
[c,1]=wavedec(S2014enero1,2,'haar');
[cd1,cd2]=detcoef(c,1,[1 2]);
a2=appcoef(c,1,'haar');
%Nombraremos las variables recompuestas con un 1 al final para
diferenciarlas de los coeficientes del año 2015 que no tocaremos
cd1r1=wrcoef('d',c,1,'haar',1);
cd2r1=wrcoef('d',c,1,'haar',2);
a2r1=wrcoef('a',c,1,'haar',2);
%Segundo día
[c,1]=wavedec(S2014enero8,2,'haar');
[cd1,cd2]=detcoef(c,1,[1 2]);
a2=appcoef(c,1,'haar');
cd1r2=wrcoef('d',c,1,'haar',1);
cd2r2=wrcoef('d',c,1,'haar',2);
a2r2=wrcoef('a',c,1,'haar',2);
%Tercer día
[c,1]=wavedec(S2014enero15,2,'haar');
[cd1,cd2]=detcoef(c,1,[1 2]);
a2=appcoef(c,1,'haar');
cd1r3=wrcoef('d',c,1,'haar',1);
cd2r3=wrcoef('d',c,1,'haar',2);
a2r3=wrcoef('a',c,1,'haar',2);
%Cuarto día
[c,1]=wavedec(S2014enero22,2,'haar');
[cd1,cd2]=detcoef(c,1,[1 2]);
a2=appcoef(c,1,'haar');
cd1r4=wrcoef('d',c,1,'haar',1);
cd2r4=wrcoef('d',c,1,'haar',2);
a2r4=wrcoef('a',c,1,'haar',2);
%Calculamos las distancias de estos días con el miércoles de 2015.
Calcularemos las distancias de cada coeficiente por separado con su
correspondiente del 2015.
```

```

d11=sum((cd1r-cd1r1).^2);
d12=sum((cd2r-cd2r1).^2);
d13=sum((a2r-a2r1).^2);
%2n día
d21=sum((cd1r-cd1r2).^2);
d22=sum((cd2r-cd2r2).^2);
d23=sum((a2r-a2r2).^2);
%3r día
d31=sum((cd1r-cd1r3).^2);
d32=sum((cd2r-cd2r3).^2);
d33=sum((a2r-a2r3).^2);
%4t día
d41=sum((cd1r-cd1r4).^2);
d42=sum((cd2r-cd2r4).^2);
d43=sum((a2r-a2r4).^2);
%creamos un vector referido a las distancias de cada nivel (cd1, cd2 y
a2) que usaremos para comparar. También vectores que guarden los
coeficientes y la semana del año a la que pertenecen
distcd1=[d11,d21,d31,d41];
distcd2=[d12,d22,d32,d42];
dista2=[d13,d23,d33,d43];
v1=[cd1r1,cd1r2,cd1r3,cd1r4];
v2=[cd2r1,cd2r2,cd2r3,cd2r4];
v3=[a2r1,a2r2,a2r3,a2r4];
%s1 referido a cd1, s2 referido a cd2r y s3 para los a2r
s1=[1,2,3,4]; s2=[1,2,3,4]; s3=[1,2,3,4];
%Hacemos copias de estos vectores
copiadistcd1=distcd1;
copiadistcd2=distcd2;
copiadista2=dista2;
copiav1=v1;
copiav2=v2;
copiav3=v3;
copias1=s1;
copias2=s2;
copias3=s3;
%Ordenamos de más cercano a más lejano las distancias de los días que
tenemos con el de 2015
distcd1=sort(distcd1,'ascend');
distcd2=sort(distcd2,'ascend');
dista2=sort(dista2,'ascend');
%Ordenamos los vectores 'v' y 's' igual que los 'dist' que acabamos de
hacer
for n=1:4
    if distcd1(n)==copiadistcd1(n)
        s1(n)=copias1(n);
        v1(:,n)=copiav1(:,n);
        n=n+1;
    else
        if n<4 & distcd1(n)==copiadistcd1(n+1)
            s1(n)=copias1(n+1);
            v1(:,n)=copiav1(:,n+1);
        else
            if n<3 & distcd1(n)==copiadistcd1(n+2)
                s1(n)=copias1(n+2);
                v1(:,n)=copiav1(:,n+2);
            else
                if n<2 & distcd1(n)==copiadistcd1(n+3)

```

```

        s1(n)=copias1(n+3);
        v1(:,n)=copiav1(:,n+3);
    else
        if n>1 & distcd1(n)==copiadistcd1(n-1)
            s1(n)=copias1(n-1);
            v1(:,n)=copiav1(:,n-1);
        else
            if n>2 & distcd1(n)==copiadistcd1(n-2)
                s1(n)=copias1(n-2);
                v1(:,n)=copiav1(:,n-2);
            else
                if n>3 & distcd1(n)==copiadistcd1(n-3)
                    s1(n)=copias1(n-3);
                    v1(:,n)=copiav1(:,n-3);
                end
            end
        end
    end
end
end
end
end
n=n+1;
end
%asignamos las variables a los nuevos coeficientes ordenados.
d11=distcd1(1);
d21=distcd1(2);
d31=distcd1(3);
d41=distcd1(4);
cd1r1=v1(:,1);
cd1r2=v1(:,2);
cd1r3=v1(:,3);
cd1r4=v1(:,4);
%Hacemos lo mismo con los coeficientes cd2
for k=1:4
    if distcd2(k)==copiadistcd2(k)
        s2(k)=copias2(k);
        v2(:,k)=copiav2(:,k);
        k=k+1;
    else
        if k<4 & distcd2(k)==copiadistcd2(k+1)
            s2(k)=copias2(k+1);
            v2(:,k)=copiav2(:,k+1);
        else
            if k<3 & distcd2(k)==copiadistcd2(k+2)
                s2(k)=copias2(k+2);
                v2(:,k)=copiav2(:,k+2);
            else
                if k<2 & distcd2(k)==copiadistcd2(k+3)
                    s2(k)=copias2(k+3);
                    v2(:,k)=copiav2(:,k+3);
                else
                    if k>1 & distcd2(k)==copiadistcd2(k-1)
                        s2(k)=copias2(k-1);
                        v2(:,k)=copiav2(:,k-1);
                    else
                        if k>2 & distcd2(k)==copiadistcd2(k-2)
                            s2(k)=copias2(k-2);
                            v2(:,k)=copiav2(:,k-2);
                        end
                    end
                end
            end
        end
    end
end

```

```

else
    if k>3 & distcd2(k)==copiadistcd2(k-3)
        s2(k)=copias2(k-3);
        v2(:,k)=copiav2(:,k-3);
    end
end
end
end
end
end
end
end
k=k+1;
end
d12=distcd2(1);
d22=distcd2(2);
d32=distcd2(3);
d42=distcd2(4);
cd2r1=v2(:,1);
cd2r2=v2(:,2);
cd2r3=v2(:,3);
cd2r4=v2(:,4);
%para a2
for t=1:4
    if dista2(t)==copiadista2(t)
        s3(t)=copias3(t);
        v3(:,t)=copiav3(:,t);
        t=t+1;
    else
        if t<4 & dista2(t)==copiadista2(t+1)
            s3(t)=copias3(t+1);
            v3(:,t)=copiav3(:,t+1);
        else
            if t<3 & dista2(t)==copiadista2(t+2)
                s3(t)=copias3(t+2);
                v3(:,t)=copiav3(:,t+2);
            else
                if n<2 & dista2(t)==copiadista2(t+3)
                    s3(t)=copias3(t+3);
                    v3(:,t)=copiav3(:,t+3);
                else
                    if t>1 & dista2(t)==copiadista2(t-1)
                        s3(t)=copias3(t-1);
                        v3(:,t)=copiav3(:,t-1);
                    else
                        if t>2 & dista2(t)==copiadista2(t-2)
                            s3(t)=copias3(t-2);
                            v3(:,t)=copiav3(:,t-2);
                        else
                            if t>3 & dista2(t)==copiadista2(t-3)
                                s3(t)=copias3(t-3);
                                v3(:,t)=copiav3(:,t-3);
                            end
                        end
                    end
                end
            end
        end
    end
end
end
end
end
end

```

```

t=t+1;
end
d13=dista2(1);
d23=dista2(2);
d33=dista2(3);
d43=dista2(4);
a2r1=v3(:,1);
a2r2=v3(:,2);
a2r3=v3(:,3);
a2r4=v3(:,4);
%creación y comparación del resto de semanas del año
for m=5:52
    dia=(m-1)*7+1; %(con esto conseguimos el día que queremos) (m=semana)
    p=(dia-1)*96+1;
    q=p+95;
    x=S2014(p:q);
    %descomponemos y recomponemos por separado
    [c,l]=wavedec(x,2,'haar');
    [cd1,cd2]=detcoef(c,l,[1 2]);
    a2=appcoef(c,l,'haar');
    %ponemos cd1r5 para que no haya confusión con los 4 anteriores
    cd1r5=wrcoef('d',c,l,'haar',1);
    cd2r5=wrcoef('d',c,l,'haar',2);
    a2r5=wrcoef('a',c,l,'haar',2);
    %Calculamos las distancias de este día con el de 2015.
    d51=sum((cd1r-cd1r5).^2);
    d52=sum((cd2r-cd2r5).^2);
    d53=sum((a2r-a2r5).^2);
    %toca comparar
    %ya está ordenado, así que haremos un filtro de que, si la nueva
    distancia es mayor que la distancia en la posición 4, ya no comparemos
    más y descartamos esta semana.
    %para cd1
    if d51>d41
        m=m+1;
    else
        %si es menor que el 4t valor, lo sustituimos y volvemos a ordenar
        los vectores
        s1(4)=m;
        cd1r4=cd1r5;
        d41=d51;
        distcd1=[d11,d21,d31,d41];
        v1=[cd1r1,cd1r2,cd1r3,cd1r4];
        copiadistcd1=distcd1;
        copiav1=v1;
        copias1=s1;
        distcd1=sort(distcd1,'ascend');
        for n=1:4
            if distcd1(n)==copiadistcd1(n)
                s1(n)=copias1(n);
                v1(:,n)=copiav1(:,n);
                n=n+1;
            else
                if n<4 & distcd1(n)==copiadistcd1(n+1)
                    s1(n)=copias1(n+1);
                    v1(:,n)=copiav1(:,n+1);
                else
                    if n<3 distcd1(n)==copiadistcd1(n+2)

```

3)

```

        s1(n)=copias1(n+2);
        v1(:,n)=copiav1(:,n+2);
    else
        if n<2 & distcd1(n)==copiadistcd1(n+3)
            s1(n)=copias1(n+3);
            v1(:,n)=copiav1(:,n+3);
        else
            if n>1 & distcd1(n)==copiadistcd1(n-1)
                s1(n)=copias1(n-1);
                v1(:,n)=copiav1(:,n-1);
            else
                if n>2 & distcd1(n)==copiadistcd1(n-2)
                    s1(n)=copias1(n-2);
                    v1(:,n)=copiav1(:,n-2);
                else
                    if n>3 & distcd1(n)==copiadistcd1(n-
                        s1(n)=copias1(n-3);
                        v1(:,n)=copiav1(:,n-3);
                    end
                end
            end
        end
    end
end
end
end
end
end
n=n+1;
end
end
d11=distcd1(1);
d21=distcd1(2);
d31=distcd1(3);
d41=distcd1(4);
cd1r1=v1(:,1);
cd1r2=v1(:,2);
cd1r3=v1(:,3);
cd1r4=v1(:,4);
%para cd2
if d52<d42
    s2(4)=m;
    cd2r4=cd2r5;
    d42=d52;
    distcd2=[d12,d22,d32,d42];
    v2=[cd2r1,cd2r2,cd2r3,cd2r4];
    copiadistcd2=distcd2;
    copiav2=v2;
    copias2=s2;
    distcd2=sort(distcd2,'ascend');
    for k=1:4
        if distcd2(k)==copiadistcd2(k)
            s2(k)=copias2(k);
            v2(:,k)=copiav2(:,k);
            k=k+1;
        else
            if k<4 & distcd2(k)==copiadistcd2(k+1)
                s2(k)=copias2(k+1);
                v2(:,k)=copiav2(:,k+1);
            else

```

```

        if k<3 distcd2(k)==copiadistcd2(k+2)
            s2(k)=copias2(k+2);
            v2(:,k)=copiav2(:,k+2);
        else
            if k<2 & distcd2(k)==copiadistcd2(k+3)
                s2(k)=copias2(k+3);
                v2(:,k)=copiav2(:,k+3);
            else
                if k>1 & distcd2(k)==copiadistcd2(k-1)
                    s2(k)=copias2(k-1);
                    v2(:,k)=copiav2(:,k-1);
                else
                    if k>2 & distcd2(k)==copiadistcd2(k-2)
                        s2(k)=copias2(k-2);
                        v2(:,k)=copiav2(:,k-2);
                    else
                        if k>3 & distcd2(k)==copiadistcd2(k-3)
                            s2(k)=copias2(k-3);
                            v2(:,k)=copiav2(:,k-3);
                        end
                    end
                end
            end
        end
    end
    k=k+1;
end
end
d12=distcd2(1);
d22=distcd2(2);
d32=distcd2(3);
d42=distcd2(4);
cd2r1=v2(:,1);
cd2r2=v2(:,2);
cd2r3=v2(:,3);
cd2r4=v2(:,4);
%para a2
if d53<d43
    s3(4)=m;
    a2r4=a2r5;
    d43=d53;
    dista2=[d13,d23,d33,d43];
    v3=[a2r1,a2r2,a2r3,a2r4];
    copiadista2=dista2;
    copiav3=v3;
    copias3=s3;
    dista2=sort(dista2,'ascend');
    for t=1:4
        if dista2(t)==copiadista2(t)
            s3(t)=copias3(t);
            v3(:,t)=copiav3(:,t);
            t=t+1;
        else
            if t<4 & dista2(t)==copiadista2(t+1)
                s3(t)=copias3(t+1);
                v3(:,t)=copiav3(:,t+1);
            end
        end
    end
end

```

```

else
    if t<3 dista2(t)==copiadista2(t+2)
        s3(t)=copias3(t+2);
        v3(:,t)=copiav3(:,t+2);
    else
        if n<2 & dista2(t)==copiadista2(t+3)
            s3(t)=copias3(t+3);
            v3(:,t)=copiav3(:,t+3);
        else
            if t>1 & dista2(t)==copiadista2(t-1)
                s3(t)=copias3(t-1);
                v3(:,t)=copiav3(:,t-1);
            else
                if t>2 & dista2(t)==copiadista2(t-2)
                    s3(t)=copias3(t-2);
                    v3(:,t)=copiav3(:,t-2);
                else
                    if t>3 & dista2(t)==copiadista2(t-3)
                        s3(t)=copias3(t-3);
                        v3(:,t)=copiav3(:,t-3);
                    end
                end
            end
        end
    end
end
end
t=t+1;
end

d13=dista2(1);
d23=dista2(2);
d33=dista2(3);
d43=dista2(4);
a2r1=v3(:,1);
a2r2=v3(:,2);
a2r3=v3(:,3);
a2r4=v3(:,4);
m=m+1;
end
semanascd1=copias1;
semanascd2=copias2;
semanasa2=copias3;
%creamos las variables del jueves y las descomponemos y nos quedamos con
los coeficientes que nos interesan en las variables juevescd1, juevescd2
y juevesa2 (las creamos previo al bucle)
juevescd1=zeros(96,4);
juevescd2=zeros(96,4);
juevesa2=zeros(96,4);
%primer bucle para cd1
for r=1:4
    p=((semanascd1(r)-1)*7+2)-1*96+1;
    q=p+95;
    jueves1=S2014(p:q);
    [c,l]=wavedec(jueves1,2, 'haar');
    [cd1,cd2]=detcoef(c,l,[1 2]);
    cd1r=wrcoef('d',c,l,'haar',1);
    juevescd1(:,r)=cd1r;

```



```

        r=r+1;
    end
    %cd2
    for w=1:4
        p=((semanascd1(w)-1)*7+2)-1)*96+1;
        q=p+95;
        jueves2=S2014(p:q);
        [c,l]=wavedec(jueves2,2, 'haar');
        [cd1,cd2]=detcoef(c,l,[1 2]);
        cd2r=wrcoef('d',c,l,'haar',2);
        juevescd2(:,w)=cd2r;
        w=w+1;
    end
    %a2
    for f=1:4
        p=((semanascd1(f)-1)*7+2)-1)*96+1;
        q=p+95;
        jueves3=S2014(p:q);
        [c,l]=wavedec(jueves3,2, 'haar');
        a2=appcoef(c,l,'haar');
        a2r=wrcoef('a',c,l,'haar',2);
        juevesa2(:,f)=a2r;
        f=f+1;
    end
    %Hacemos la media de los coeficientes de los cuatro días obtenidos
    cd1final=sum(juevescd1,2)/4;
    cd2final=sum(juevescd2,2)/4;
    a2final=sum(juevesa2,2)/4;
    %Recomponemos la señal y ya tenemos nuestra predicción
    S2015enero15prediccion=cd1final+cd2final+a2final;
    plot(S2015enero15); hold on; plot(S2015enero15prediccion)
    title('15 enero 2015');
    ylabel('kWh');
    legend('Real', 'Predicción');
    %Mostramos el valor eficaz del consumo
    rmsreal=rms(S2015enero15);
    rmsprediccion=rms(S2015enero15prediccion);
    %aquí mostrará en pantalla los vectores semana
    Tsemanas=table(semanascd1,semanascd2,semanasa2,rmsreal,rmsprediccion)

```

